

PROJECT ADMINISTRATION DATA SHEET

Project No. E-21-632

ORIGINAL



REVISION NO. _____

GTRI/GHT

DATE 9 / 13 / 84Project Director: Dr. Phillip E. AllenSchool/Lab EESponsor: Semiconductor Research CorporationType Agreement: Contract No. 84-07-051Award Period: From 9/1/84 To 8/31/85 (Performance) 11/30/85 (Reports)

Sponsor Amount:

This ChangeTotal to DateEstimated: \$ 100,638\$ 100,638Funded: \$ 100,638\$ 100,638

Cost Sharing Amount: \$ _____ Cost Sharing No: _____

Title: A multilevel Analog Integrated Circuit CAD Program

ADMINISTRATIVE DATA

OCA Contact

Brian J. LindbergX4820

1) Sponsor Technical Contact:

2) Sponsor Admin/Contractual Matters:

* Dr. Ralph CavinRichard LaScalaSemiconductor Research CorporationContract Administrator300 Park Dr.Semiconductor Research CorporationP.O. Box 12053300 Park Dr.Research Triangle Park, NC 27709P.O. Box 12053Research Triangle Park, NC 27709Defense Priority Rating: N/AMilitary Security Classification: N/A(or) Company/Industrial Proprietary: N/A

RESTRICTIONS

See Attached N/A Supplemental Information Sheet for Additional Requirements.

Travel: Foreign travel must have prior approval – Contact OCA in each case. Domestic travel requires sponsor approval where total will exceed greater of \$500 or 125% of approved proposal budget category.

Equipment: Title vests with Sponsor for all equipment with value of \$1,000 or more.

COMMENTS:

Continuing award, anticipated for two more years.

COPIES TO:

I.D. No. 02-219-000-84-005

Project Director
Research Administrative Network
Research Property Management
AccountingProcurement/EES Supply Services
Research Security Services
Reports Coordinator (OCA)
Research Communications (2)GTRI
Library
Project File
Other I. Newton

SPONSORED PROJECT TERMINATION/CLOSEOUT SHEET

Date 11/20/86

Project No. E-21-632 School/~~LXX~~ EE

Includes Subproject No.(s) N/A

Project Director(s) Dr. Phillip E. Allen GTRC / ~~SKK~~

Sponsor Semiconductor Research Corporation

Title A multilevel Analog Integrated Circuit CAD Program

Effective Completion Date: 8/31/86 (Performance) 11/30/86 (Reports)

Grant/Contract Closeout Actions Remaining:

- ☐ None
- ☒ Final Invoice or Final Fiscal Report
- ☐ Closing Documents
- ☐ Final Report of Inventions
- ☐ Govt. Property Inventory & Related Certificate
- ☐ Classified Material Certificate
- ☐ Other _____

Continues Project No. _____ Continued by Project No. E21-F10

COPIES TO:

Project Director
Research Administrative Network
Research Property Management
Accounting
Procurement/GTRI Supply Services
Research Security Services
Reports Coordinator (OCA)
Legal Services

Library
GTRC
Research Communications (2)
Project File
Other I. Newton
A. Jones
R. Embry



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

EPHONE: (404) 894- 6251

MEMORANDUM

December 17, 1984

TO: Ralph K. Cavin
FROM: Phillip E. Allen
SUBJECT: Quarterly Report, Sept. 1, 1984 - Dec. 1, 1984, SRC
Proj. 84-07-051

INTRODUCTION

This Quarterly Report covers the period from Sept. 1, 1984, to Dec. 1, 1984, for the SRC sponsored research program at Georgia Institute of Technology titled "A Multilevel Analog Integrated Circuit CAD Program". The objective of this research program is to develop and implement a multilevel analog IC design program which will increase analog IC design productivity, insure testability, and to be compatible with CAD techniques for digital IC design.

The primary activity during this quarter has been the transferring of this research program, which was sponsored by SRC at Texas A&M University from April 1, 1983, until August 30, 1984, to Georgia Institute of Technology. Although the supported research is now at Georgia Institute of Technology, the research results and interests between Texas A&M University and Georgia Institute of Technology remain closely linked. The program at Georgia Institute of Technology consists of 5 students (3 MS and 2 Ph'd), two of which are supported by SRC. Since research was done both at Texas A&M University and Georgia Institute of Technology during this period, it will be described separately in the following.

RESEARCH ACTIVITIES AT TEXAS A&M UNIVERSITY

Research activity continued at Texas A&M University during this Quarter because it was also part of theses research which was not completed on August 30, 1984. The MS research of two students, Steve Bily and Edmond Macaluso, were involved in the SRC research at Texas A&M University. Mr. Bily's research included the development, application, and testing of the prototype AIDE program. His MS thesis should be complete by January 1985, and has been delayed primarily by extensive experimental data which he has made. The title of Mr. Bily's MS thesis is "A Functional Cell Design Methodology for Analog Integrated Circuits".

The MS research of Mr. Macaluso was devoted entirely to the development and foundation of the AIDE 2 program. Figure 1 shows the overall concept of the AIDE 2 program and its use as a toolkit. He established the data transfer structures, the procedural language and format, and the general protocol of the program. This research is described in detail in his thesis titled "A Procedural Language for Automated Analog Circuit Design", which was completed in November, 1984. A copy of this thesis will be forwarded to SRC. A short paper presentation titled "An Automated Analog CAD Program" was given at the 1984 ICCAD. (Short papers were not published.)

While finishing up his MS research, Mr. Macaluso wrote a routine which connected the AROMA program with AIDE2 allowing the automated design of a class of SC filters. AROMA is a high level design program for cascaded filters developed at Texas A&M University by Dr. Edgar Sanchez-Sinencio. This provided a tool which produced a layout and simulation of a cascade SC filter from a high level description of the

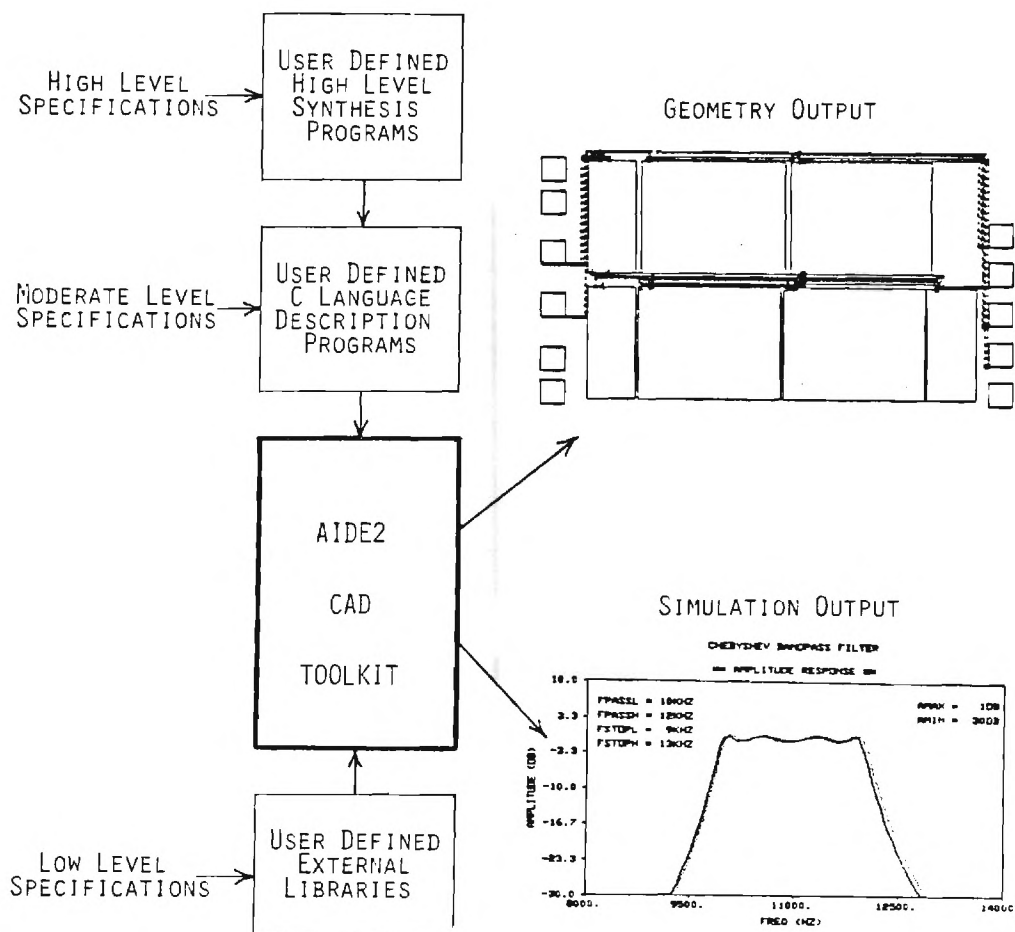
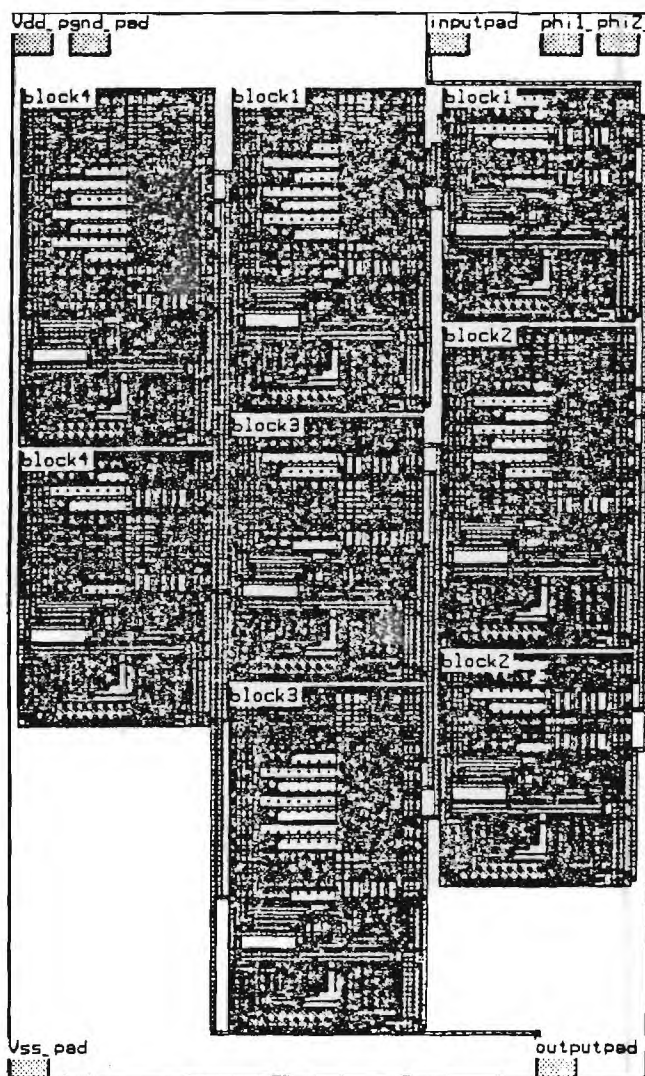


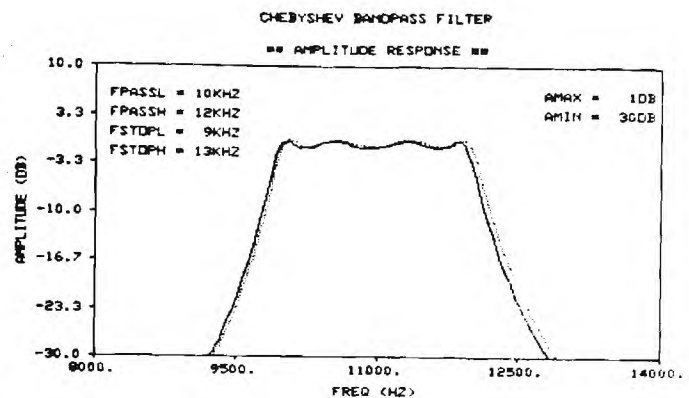
Fig. 1 - Illustration of AIDE2 as a toolkit for analog IC design.

filter. Figure 2 shows the layout and simulation from this CAD tool for an 8th order, bandpass SC filter. A reprint of a paper presented at ICCAD 84 is included as Attachment A and describes the program in more detail.

Other work at Texas A&M University includes the development of a linear analog summer/integrator block as the first entry into the AIDE 2 library. Simultaneously, the means of simulating the linear analog block using the SPICE-PAC program was developed. This work is almost complete and will be incorporated into AIDE2 at that time.



(a)



(b)

Fig. 2 - (a) Layout and (b) Simulation of an 8th order bandpass filter.

RESEARCH ACTIVITIES AT GEORGIA INSTITUTE OF TECHNOLOGY

The primary activity in this research program at Georgia Institute of Technology has been to develop the facilities and recruit personnel for this research. On October 1984, a VAX 11/780 with UNIX 4.2 was installed and brought up. This machine serves only the microelectronics program and a portion of the digital signal processing programs and provides more than sufficient computing capability for the SRC research. The VLSI CMOS software tools from the University of

Washington/Northwest Consortium have been acquired and have been compiled. Terminals for the VAX have been placed in the graduate student offices facilitating their SRC research. Four VT240C terminals, two AED 767 terminals, one Printronix and one HP plotter have been ordered to increase the I/O capability and to use the VLSI software. Figure 3 shows the microelectronics VAX computer facility. Also

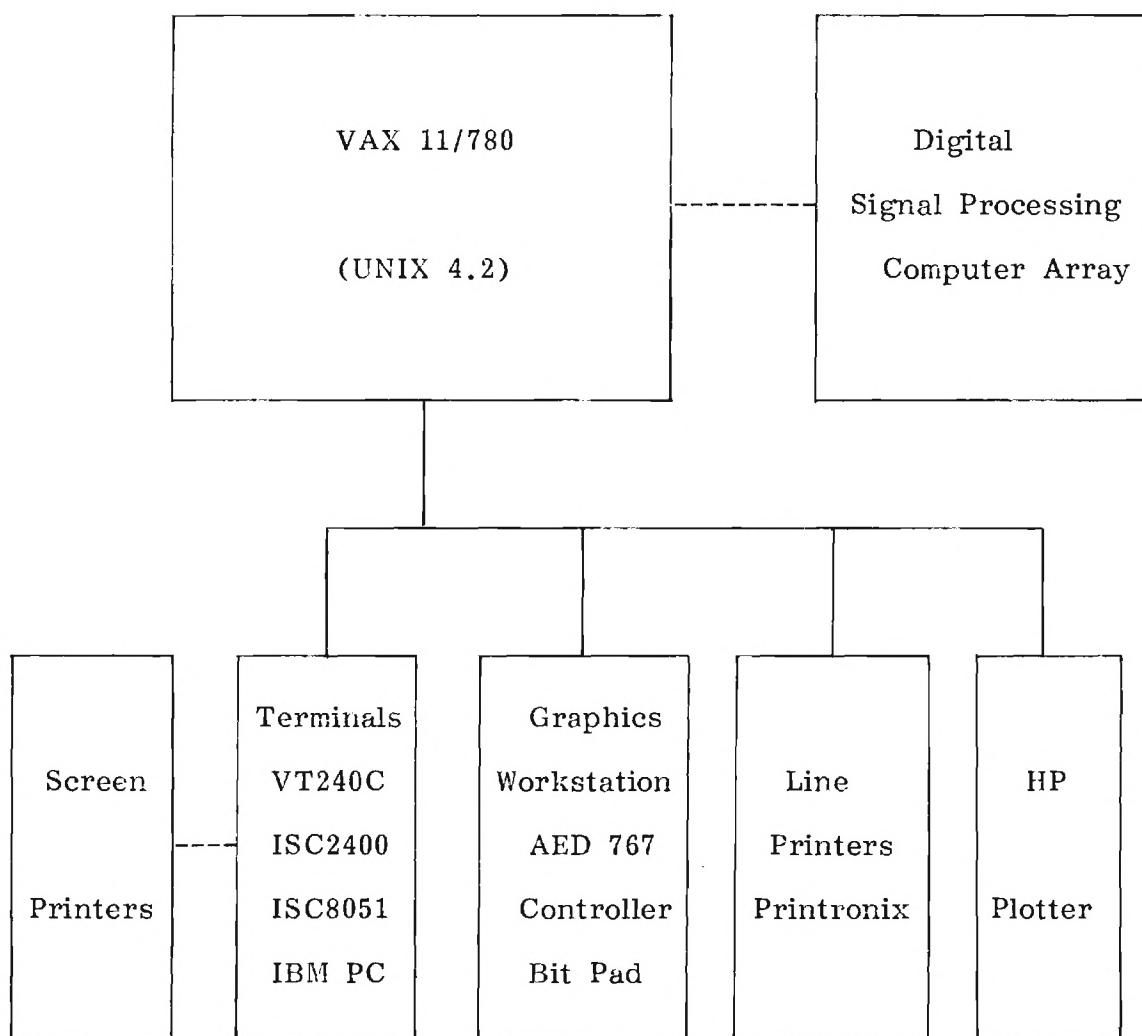


Fig. 3 - Computer resource configuration for the microelectronic program.

available are 2 CALMA workstations, two Harris 800's, and a CYBER. A proposal to Hewlett Packard has been made to implement an IC test facility at Georgia Institute of Technology to make experimental measurements and diagnosis of the integrated circuits that we will design and have fabricated at Texas Instruments and Harris Semiconductor. Fig. 4 shows the planned IC test facility.

The first task was to find qualified students to participate in this research program. Two graduate students (Pat Barton and Bruce Young) were former Texas A&M University students and were familiar with the research. These two students were supported by Georgia Institute of Technology. Three other students were hired to work on the SRC research. Two of these students, Raouf Halim (MS) and Kwang Yoon (Ph'D), are supported by SRC and a third, Seong Hong (Ph'D), is supported by Schlumberger funds.

AIDE2

During the early part of November, Edmond Macaluso brought the latest version of AIDE2 to Georgia Institute of Technology, installed it in our VAX and demonstrated how to use it. Our present efforts are centered around debugging, expanding and documenting AIDE2, investigating the problem of technology independence, and developing the concepts of testability for analog IC designs. Allied research efforts (not supported by SRC) include the development of a high level design/synthesis program for successive approximation A/D converters which will interface to AIDE2 to form a compiler for a class of A/D converters and the implementation of our analog CAD tools on advanced PC hardware.

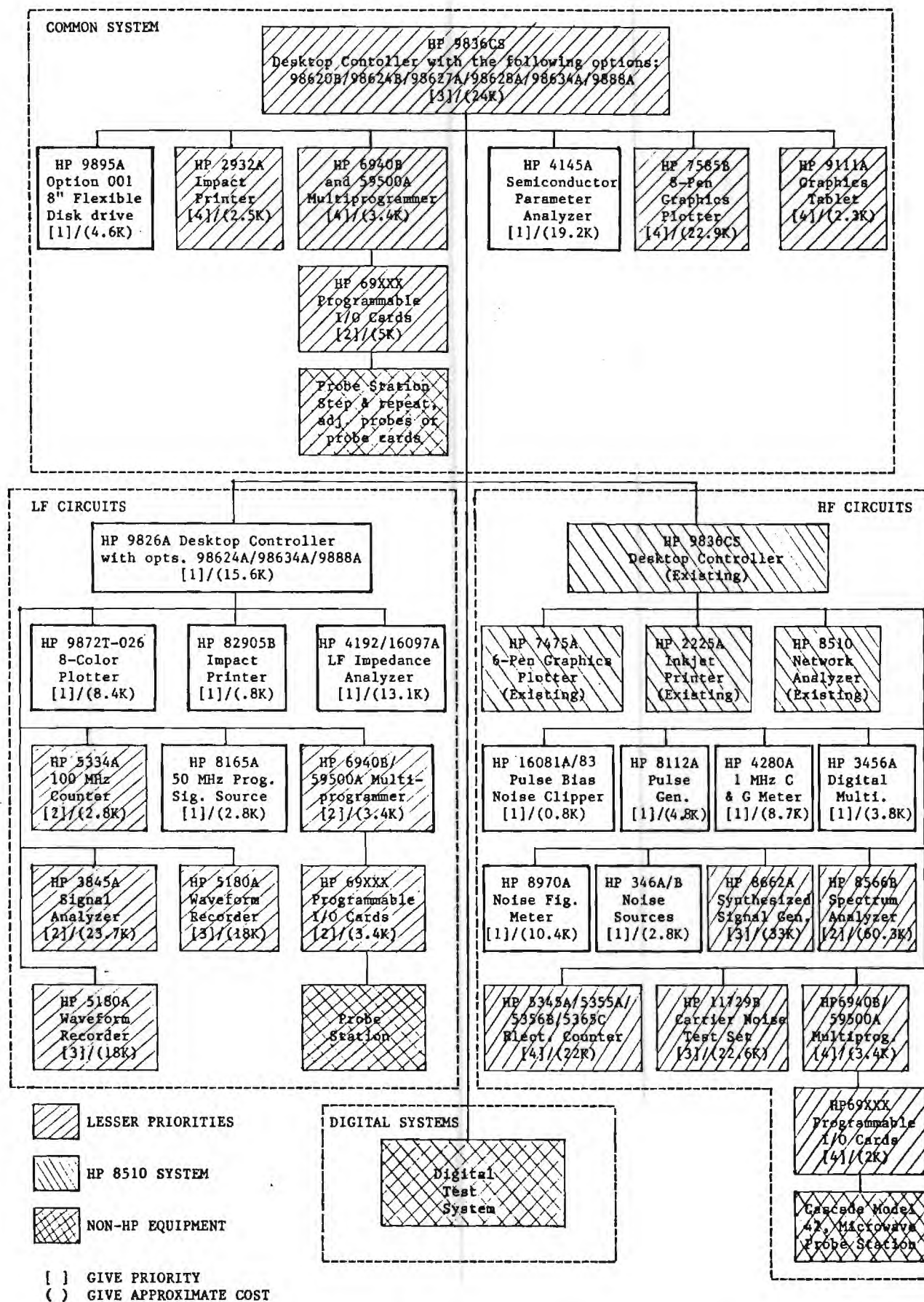


Fig. 4 - Proposed IC test facility configuration

The present plans and schedule of activities for future efforts are illustrated in Figure 5. This schedule covers a period of three years and assumes that SRC will continue to fund this research. The primary effort is the development of the AIDE2 program from the AIDE2.0 through AIDE2.3 versions. AIDE2.0 is the present status of AIDE2. The basic concepts and foundation of the toolkit have been established. It has not yet been used to generate designs for silicon and needs to have the simulation capability added. The cell library presently has only a linear, summing, amplifier/integrator. AIDE2.0 will be used to design SC filters to be fabricated at TI.

AIDE2.1 will be completed in September 1985. Its major feature will be technology independence. The software should be able to work with a limited range of silicon gate, CMOS technologies. The projected technology range is from 3 to 5 microns for an n-well, p-well, or twin well CMOS technology. AIDE2.1 will also have other blocks added to its cell library, including a clock multiplexer and a comparator. AIDE2.1 will be released to interested SRC industrial test sites.

AIDE2.2 will be completed in September, 1986, and will incorporate the testability and higher level simulation concepts resulting from the previous research. AIDE2.2 will also interface with a high level synthesis program for designing successive approximation A/D converters. It is also expected that a version of AIDE2 will be available on the IBM PC AT.

AIDE2.3 will be released on September, 1987, and will incorporate the feedback from external and internal users. The cell library is expected to have a large number of cells covering a wide range of analog circuit applications. Several high level design programs for various classes of analog circuits are expected to be a part of AIDE2.3.

Technology-Independence

The technology independence research is being conducted by Kwang Sub Yoon (Ph'D) and is progressing on schedule. A set of generic parameters which may be used to develop the simulation and geometric databases has been identified. The generic parameters must meet the following criteria:

1. Be used to achieve all SPICE model level parameters.
2. Provide all geometric parameters for the AIDE2 program.
3. Can be measured from a test vehicle.

Figure 6 shows the relationship between the various CMOS technologies, the generic parameters set, and the databases in AIDE2. The generic parameters can be categorized into process-defined parameters and user-defined parameters. Process-defined parameters include doping levels, interlayer capacitance, surface mobility, junction depth, lateral diffusion, oxide charge density and sheet resistance. User-defined parameters include area of the source and drain, periphery of the source and drain, channel width and length, temperature, and the minimum physical resolution unit (λ). Methods of measuring most of the process-defined parameters from a test vehicle have been identified and theoretically confirmed. These methods include MOSFET capacitance measurement at high frequency, reverse bias junction capacitance measurements, cross-bridge resistance measurement and channel transconductance measurement.

During the next quarter a reliable test method and test structure will be developed for lateral diffusion length and the n/p poly doping levels. A major effect of the next quarter will be (1) to assemble each sub-test structure into a complete test structure which can permit the measurement of all generic parameters, (2) to develop a program to

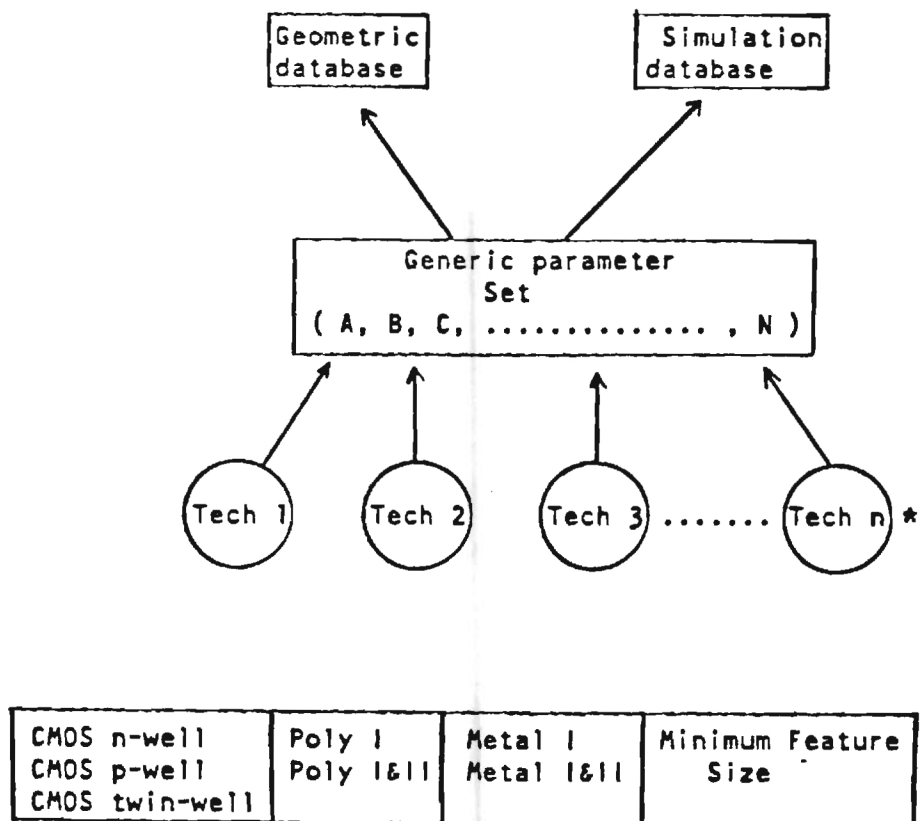


Fig. 6 - Representation of the generic parameter set.

generate the simulation and geometric parameters from the measured generic parameters, and (3) to investigate the choice of generic parameters for other technologies as the opportunity exists.

A problem which has not been solved satisfactorily is a generic design rule set. One approach which is attractive is to adopt Griswold's approach of a CMOS design rule set which is sufficiently relaxed so that it will accommodate all technologies [1]. Another approach may be to sub-classify technologies in order to be more efficient in area usage. These questions will be considered in more detail next quarter.

Testability for Analog IC's

Research in the area of testability for analog IC's is being conducted by Seong-Kwan Hong (Ph'D). This research has been divided into several phases. The first is to be able to physically access/isolate parts of an analog block, subsystem, or system. The second is to be able to self diagnose a fault and the third is to be able to isolate the fault through redundancy. Since the concept of "fault" is not well defined for analog circuits, the second and third phases face some challenging tasks. The research for the first year or so will be focused on the first phase which will be called testability.

A test structure will be incorporated into AIDE2 which should meet the following requirements. It should be independent of the contents of the analog block and the number of input and output ports. It should be able to guarantee that each component of the cellular or block level can be isolated from other circuits and externally tested. Finally, it

1. T.W. Griswold, "Portable Design Rules for Bulk CMOS," VLSI Design, pp. 62-66, Sept./Oct. 1982.

should maximize the capability to test combinations of blocks, i.e. subsystem and minimize the testing area.

Several possible solutions to the testability problem have been explored. There are several categories of ports to an analog block (standard cell) which have been identified. These include ports which are not tested, ports which are to be connected to either the regular routing or to an external test pad and ports which are to be connected to the regular routing and to an external test pad.

The objective of the test structure is to implement these three types of ports. In the first case nothing is required. In the second and third cases a double throw switch is necessary. The control of the switch is determined by a cascaded shift register which can be serially loaded to achieve any of the combinations as indicated in Figure 7. Depending on how the shift registers are connected to the double throw switch, the second and third test modes may be implemented. If one or

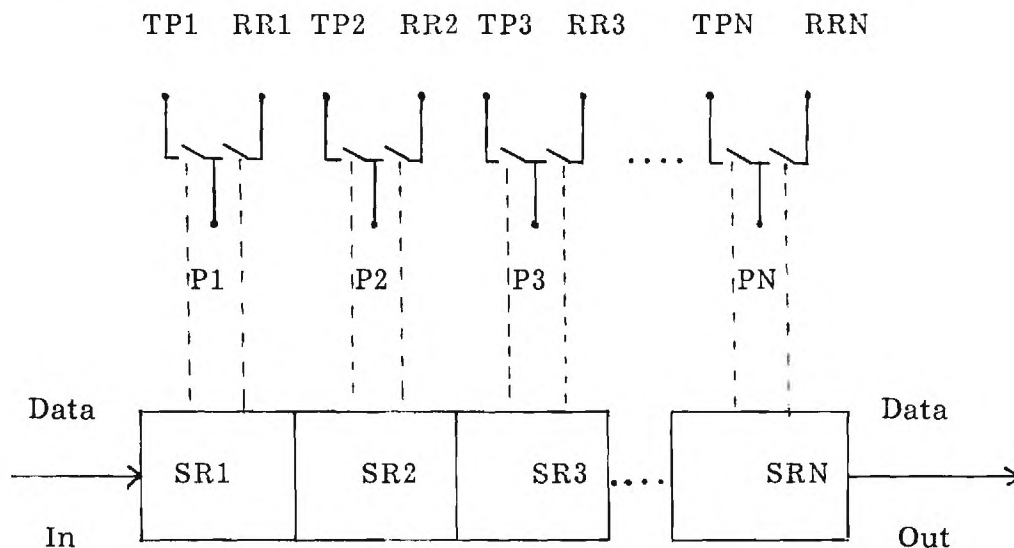


Fig. 7 - Basic concept of the test structure. TP_i is the test pad destination for the i th port, RR_i is the regular routing destination for the i th port, and P_i is the i th port. SR_i is the shift register controlling the i th double throw switch.

more ports require the same test mode, then on shift register can control multiple sets of double throw switches.

The physical implementation of the test structure has two possibilities. The first is to place the test structure between the regular routing channel and the analog standard cell row. The second is to include the test structure as a regular standard cell. The second approach is more economical of area and is illustrated in Figure 8. This topology has the advantage of being able to use the CAD tools of AIDE2 which would normally be applied to the analog blocks. Presently, the various cells of the test block, such as the shift register and double-throw switch, have been layed out and will be fabricated.

Plans for next quarter include developing the protocol and methodology for programming the test structure and implementing this in AIDE2. A large number of cases need to be examined in order to establish the optimum choices. Other considerations, such as the influence of the switches in analog signal paths, the number of external test pads and the area trade-offs, need to be re-examined. It also appears that for testability of analog circuits to be successful, a test schedule and requirements must be developed at the time the design is defined. For example, the test program could key from the simulation performed by the designer. Any simulation performed during the design should be repeatable as an external measurement when the proper data is entered into the serial shift registers.

High Level Synthesis Program for AD Converters

This research will result in a high level program which will generate a layout file and a simulation file for the successive approximation class of A/D and D/A converters. The synthesized converter will be tailored to its specific application based on the

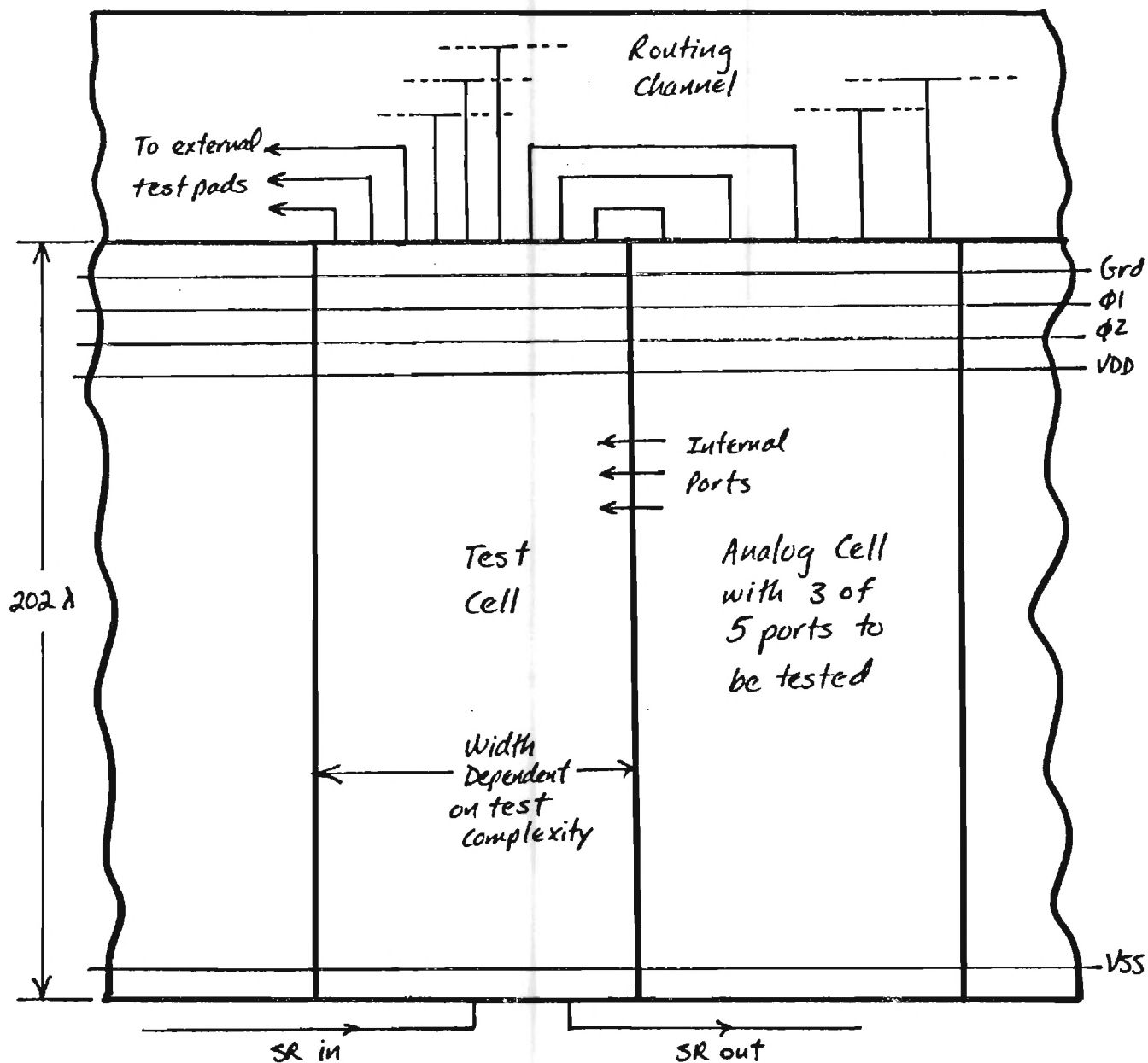


Fig. 8 - Area efficient floorplan for the testability of analog cells.

parameters input by the user. The program will generate parameterized cells which will be placed and routed by the AIDE2 program. The parameterized cells will be designed for a 5 micron n-well silicon gate process. The basic DAC topologies have been chosen, accuracy calculations performed, and capacitor layout topologies have been considered. Topologies for the DAC include capacitor arrays, resistor strings, and a combination of the two. A coupling capacitor may be used in the capacitor array to make a two-stage array, thus saving area and possibly increasing linearity.

Accuracy calculations based on device mismatch have been performed for each of the topologies but have not yet been verified. The goal of each derivation is to express the differential nonlinearity and the integral nonlinearity as a function of the percentage change in the capacitors and resistors. This will allow the program to determine the proper topology to meet design specifications.

The capacitor layout will be in a one-dimensional common centroid geometry. Switches will be included near each capacitor so the parasitics between the capacitors and switches will be minimized and independent of routing between cells.

Test chips of resistor and capacitor arrays will be submitted for fabrication in January. The MS research proposal will be written and submitted to the school early in the quarter. Parameterized cell routines will be developed for the resistor and capacitor arrays.

PERSONNEL

The personnel involved with the project are Phillip Allen (Principle Investigator and Professor), Pat Barton (MS student), Raouf Halim (MS student), Seong Hong (Ph'D student), Kwang Yoon (Ph'D student), and Bruce Young (MS student). Mrs. Sherri Brenner has been supported

part-time as a secretary. Mrs. Brenner, Mr. Halim, and Mr. Yoon have been supported by SRC during this quarter.

EXPENDITURES

The expenditures for the period of September 1, 1984, through December 1, 1984, are as follows:

	<u>9/84</u>	<u>10/84</u>	<u>11/84</u>	<u>Total</u>
Personal Services	0	\$1698.50	\$2665.45	\$4363.95
Fringe Benefits	0	\$6.74	\$68.97	\$75.71
Materials & Supplies	0	\$57.00	\$830.21	\$887.21
Computer	0	0	0	\$0.00
Overhead	<u>0</u>	<u>\$974.52</u>	<u>\$1971.24</u>	<u>\$2945.76</u>
Totals	0	\$2736.76	\$5535.87	\$8272.63

AN AREA OPTIMIZED CAD PROGRAM FOR CASCADE SC FILTER DESIGN

Edgar Sánchez-Sinencio, JongJung Lee
Texas A & M University
Dept. of Electrical Engineering
College Station, TX 77843

Phillip E. Allen
Georgia Institute of Technology
School of Electrical Engineering
Atlanta, GA 30332

Abstract.— An area optimized computer-aided cascade SC filter design program is presented. The program permits the user tradeoffs between several design parameters such as passive sensitivity, op amp output voltage swings, and the total capacitance of the filter. The program has default values of the tradeoffs, however each block in the cascade filter can be modified by the user to meet specific requirements. The program is user-oriented, it can start, in its simplest mode, with a set of frequency specifications and the output gives a circuit description file that can be laid out by an analog circuit layout generator program.

I. Introduction

Switched-capacitor (SC) filters have reached certain maturity [1 - 3] especially in voice band applications. However, the need for better performance, reasonable sensitivity values, and reduced design and chip cost is still a problem. Motivated by the above problem, the computer program AROMA was developed.

The purpose of this paper is to describe an automatic-scheme approach to design cascade SC filters. The end result of the program is a circuit description file that can be laid out by a layout generator program.

The cascade design was chosen for its simplicity and practicality. This design approach is very appropriate to filter functions of temperate stringency. In fact, cascade SC filters is a good viable design alternative chosen by several active groups [3 - 6] in monolithic filter design.

AROMA can design a filter given the frequency specifications (other alternatives which are described later). The program has an strategic algorithm to reduce the total capacitances at the expense of passive sensitivity. Additionally, the program is interactive, and allows the user the flexibility to rearrange the cascade block sequence and/or to do the pairing of poles and zeros [7]. Besides, further tradeoffs of the whole filter or particular cascaded blocks can be individually tailored by the user to satisfy specific design constraints.

II. Building Block Structures

The basic first-order SC topology is shown in Fig. 1. This structure can implement either a lowpass or a high pass function needed for odd order filters. A general basic second-order structure, biquad 1, is shown in Fig. 2. It has been shown that this structure has small gain-bandwidth product (GB) effects [8]. Furthermore, the biquadratic structure due to the combination of positive and negative feedback can handle tradeoffs between passive sensitivity and total capacitance area [9]. Another quasi-general biquadratic structure, biquad 2, reported in [8] is included in the built-in library second-order blocks. The biquad 2 has also reduce GB effects and can realize any 2nd-order filter function, except the bilinear BP. An all-pass, second-order structure [6], with reduced total capacitance, is also included in the built-in library of AROMA.

It should be emphasized that in the interactive mode of the program, for a given function (i.e., LP or BP) the user can choose one of the different transfer functions with different number of zeros. The choice of a particular transfer function depends on the design constraints where an additional tradeoff between performance and cost will have to be considered. Furthermore, AROMA cascades the blocks with the optimal clock phases to minimize the GB effects [6], [8].

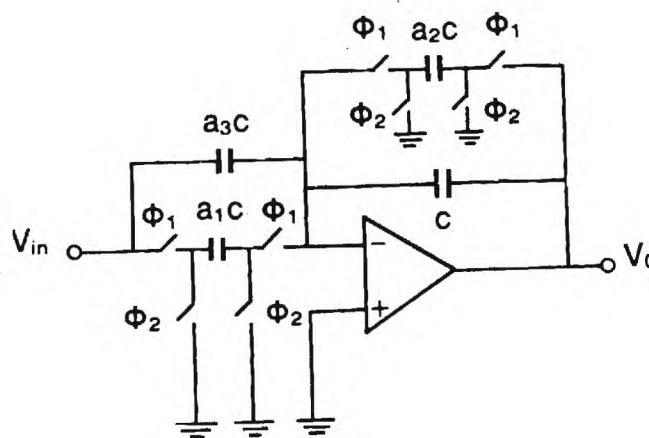


Fig. 1 First-Order SC Building Block

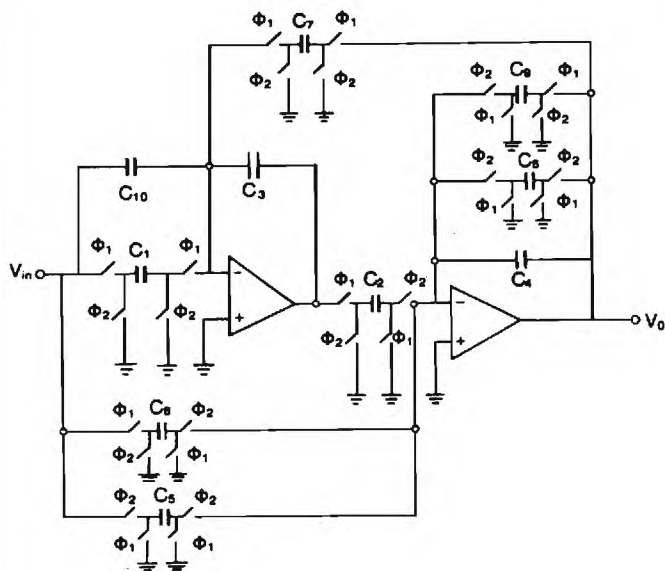


Fig. 2 A General Biquadratic SC Filter

III. Description of the Program

The basic flow chart of AROMA is shown in Fig. 3. Next we describe in some detail each of the boxes of Fig. 3. In box 1, the specification can be described by either one of the following options: i) Frequency specifications, ii) Continuous-time domain transfer function, $H(s)$, iii) Discrete-time domain transfer function, $H(z)$.

If option i) is chosen, AROMA will use an approximation technique chosen by the user to obtain $H(s)$ in cascade form, each block having a biquadratic voltage transfer function. An additional first-order section is required in the case of an order odd filter. Currently the Butterworth and Chebyshev approximations are available. In box 2, once $H(s)$ is determined, and frequency "prewarped" $H(z)$ is obtained through a bilinear mapping [1] - [2]. Now if the option ii) is chosen, $H(s)$ has to be described in cascade form and the approximation part is skipped. And if option iii) is opted both the approximation and the bilinear transformation are omitted. That is both boxes 1 and 2 are skipped.

Simultaneously with the data of box 1, the clock frequency or clock frequency range is specified by the user. In the second case the user, based on the information provided by the program, chooses the optimal clock frequency that minimizes the total capacitance area. The program to determine optimal clock frequency uses default values for the passive sensitivity, scales all the op amp output voltages to the same level, and chooses the SC topology accordingly to each individual quality factor of each cascade block. The selection of the second-order topologies is carried out in box 3. Then, the smallest capacitor associated with every op amp is individually normalized to a unit capacitance C_u and the total capacitance [1] - [2] is obtained. That is

indicated in box 4. Now if the total capacitance exceeds certain design limitations (Box 5), for instance, one of cascade block exceeds say $40C_u$ then the designer needs to adjust one or more parameters to satisfy the maximal individual cascade block capacitance area. This is indicated in box 6 where the passive sensitivity and/or maximum op amp output voltages swings are iteratively modified until a satisfactory compromise between the several parameters is achieved. If the user desires AROMA generates an input file (box 7) for the SC filter simulator SWITCAP then the program generates the output file for SWITCAP. Finally AROMA generates the layout input (box 8) file that can be coupled to a layout generator program [10].

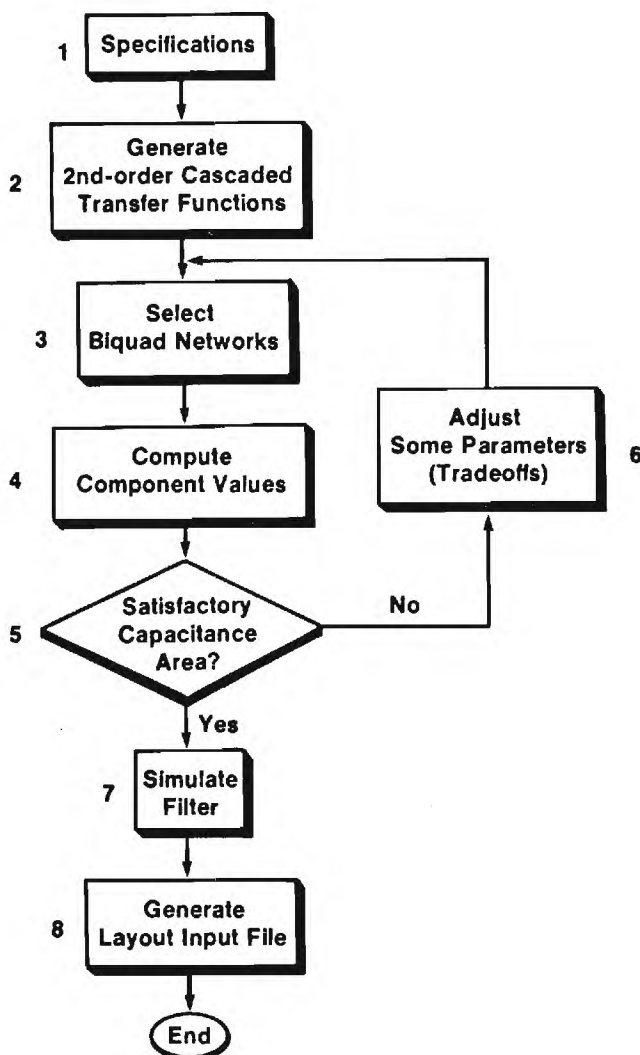


Fig. 3 Basic Flow-Diagram of AROMA

IV. An Example

As an example, the 8th-order BP Chebyshev with a 1 dB ripple is considered. The initial data is given as a set of frequency specifications. The Chebyshev approximation, $A_{min} = 30\text{dB}$, $A_{max} = 1\text{dB}$ and $f_c = 100\text{ KHz}$ are also specified by the user. AROMA designs the four 2nd-order blocks with the default value of the sensitivity of Q with respect to positive-feedback passive components as one. Furthermore all the op amp output voltages are set to a maximum of 1 volt for an input of 1 volt. Under the above conditions the total capacitance is $C_T \approx 609\text{ Cu}$. If the user modifies nonhomogeneously all the op amp output voltages to obtain an overall voltage gain of 10, the value of $C_T \approx 530\text{ Cu}$. Furthermore, if the Q -sensitivity becomes 1.5, 1.5, 3.0 and 4.0 for each 2nd order block, respectively, then $C_T \approx 302\text{ Cu}$. Nearly a 50% reduction has been obtained at the expense of Q -sensitivity and output voltage swing. The frequency response of the filter is shown in Fig. 4, where the continuous line represents $H(s)$ and the dashed line the SWITCAP response of the designed filter. More details about the component values are given in the Table 1.

Table 1a) Capacitor values for $S_G^Q = 1.0$ and maximum output $G=1$.

	S	G	TC	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
BLOCK1	1.00	1.00	88.5	0.0	33.8	20.8	21.4	1.6	0.0	6.9	2.0	1.0	1.0
BLOCK2	1.00	1.00	93.4	0.0	36.3	22.3	22.9	1.6	0.0	6.4	2.0	1.0	1.0
BLOCK3	1.00	1.00	199.8	0.0	77.3	47.5	51.7	1.6	0.0	17.7	2.0	1.0	1.0
BLOCK4	1.00	1.00	227.3	0.0	90.9	55.9	60.3	1.6	0.0	14.5	2.0	1.0	1.0
TOTAL CAPACITANCE IS 608.9 UNIT.													

Table 1b) Capacitor values for a total gain of 20dB ($G = 10$).

	S	G	TC	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
BLOCK1	1.00	1.78	77.7	0.0	33.8	11.7	21.4	2.9	0.0	3.9	2.0	1.0	1.0
BLOCK2	1.00	1.78	82.2	0.0	36.3	12.5	22.9	2.9	0.0	3.6	2.0	1.0	1.0
BLOCK3	1.00	1.78	172.5	0.0	77.3	26.7	51.7	2.9	0.0	9.9	2.0	1.0	1.0
BLOCK4	1.00	1.78	197.7	0.0	90.9	31.4	60.3	2.9	0.0	8.2	2.0	1.0	1.0
TOTAL CAPACITANCE IS 530.0 UNIT.													

Table 1c) Capacitor values for $G=10$ and different sensitivities

	S	G	TC	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
BLOCK1	1.50	1.78	57.9	0.0	22.7	11.8	13.9	1.9	0.0	3.9	1.7	1.0	1.0
BLOCK2	1.50	1.78	61.1	0.0	24.4	12.6	14.9	1.9	0.0	3.6	1.7	1.0	1.0
BLOCK3	3.00	1.78	85.5	0.0	27.0	27.0	17.2	1.0	0.0	9.9	1.4	1.0	1.0
BLOCK4	4.00	1.78	97.5	0.0	32.2	32.2	19.8	1.0	0.0	8.2	1.7	1.4	1.0
TOTAL CAPACITANCE IS 302.0 UNIT.													

V. Conclusions

AROMA is a versatile program able to deal with cascade SC filter design. It incorporates a number of recent advances such as the use of biquad structures with reduced GB effects and/or small total capacitances. Also, several tradeoffs, without affecting the response of the filter, between passive sensitivity, op amp output voltage swings and total capacitance have been implemented in the program which offers more flexibility. Significant reduced design time and chip cost are possible by using AROMA.

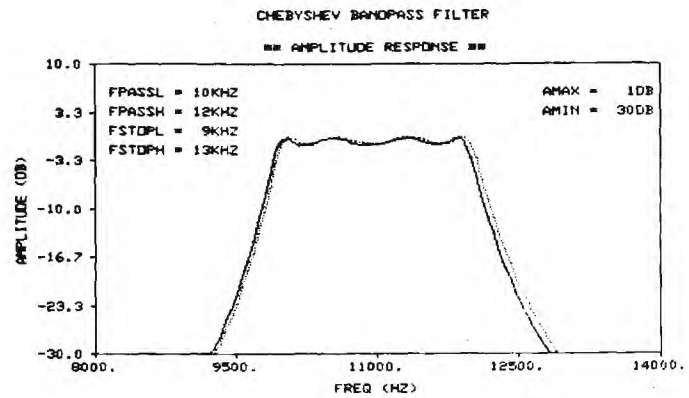


Fig. 4 Frequency Response of the Filter.

References

1. P. R. Gray, D. A. Hodges, and R. W. Brodersen, *Analog MOS Integrated Circuits*, New York: IEEE Press, 1980.
2. P. E. Allen and E. Sánchez-Sinencio, *Switched-Capacitor Circuits*, Van Nostrand Reinhold Co., 1984.
3. P. E. Fleisher, K. R. Laker, et al. "An NMOS Analog Building Block for Telecommunication Applications", *IEEE Trans. Circuits Syst.*, vol. CAS-27, pp 552-559, June 1980.
4. R. Gregorian, "Switched-Capacitor Filter Design Using Cascade Sections," *IEEE Trans. Circuits Syst.*, vol. CAS-27, pp 515-521, June 1980.
5. U. W. Brugger, D. C. Von Grünigen, and G. S. Moschytz, "A Comprehensive Procedure for the Design of Cascade Switched-Capacitor Filters," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp 803-810, August 1981.
6. K. R. Laker, A. Ganesan, and P. E. Fleisher, "Design and Implementation of Cascaded Switched Capacitor Delay Equalizers," *Proc. ISCAS/IEEE*, pp 171, Montreal, Canada, May 1984.
7. E. Lueder, "A Decomposition of a Transfer Function Minimizing Distortion and Inband Losses" *Bell Syst. Tech. J.*, vol. 49, pp 455-469, March 1970.
8. E. Sánchez-Sinencio, J. Silva-Martínez, and R. L. Geiger, "Biquadratic SC Filters with Small GB Effects" *IEEE Trans. Circuits Syst.*, vol. CAS-31 October 1984.
9. E. Sánchez-Sinencio, R. L. Geiger and J. Silva-Martínez "Tradeoffs Between Passive Sensitivity, Output Voltage Swing and Total Capacitance in Biquadratic SC Filters" *IEEE Trans. Circuits Syst.*, vol. CAS-31 November 1984.
10. P. E. Allen et al, "The Use of a CAD Tool to Design Analog CMOS Integrated Circuits," *Proc. ISCAS/IEEE*, pp 1223-1226, Montreal, Canada, May 1984.

Cindy Meyer

GEORGIA INSTITUTE OF TECHNOLOGY

School of Electrical Engineering

Atlanta, GA 30332

(404) 894-6251

March 19, 1985

Dr. Ralph K. Cavin, III
Program Manager, Design Sciences
Semiconductor Research Corporation
P.O. Box 12053
Research Triangle Park, NC 27709

Dear Ralph:

Enclosed please find two copies of the Second Quarter Report for the SRC Project No. 84-07-051. If you have any questions or need more information please do not hesitate to contact me.

I would like to schedule an SRC Program Review sometime near the middle of June if possible. My objective is to have the review before three of my people leave for the summer. During this review I would like to have a mini-seminar on how to use the AIDE2 program for those members of the SRC Industrial community who are interested. We also would like to provide copies of AIDE2 and documentation. Our objective is to try to transfer our research results and to get preliminary feedback from industrial users. I believe this is an important part of our activity if we are to be successful. I look forward to hearing from you on this matter.

I hope things are fine with you and your family. Margaret and I plan tentatively to be in your area on April 27. I don't know if you remember Tarek Ismail from TAMU or not, but he has been working for Honeywell since he graduated and was married last Christmas in Egypt. His bride lives in the RTP area and they are having a reception that we are planning to attend. Things are going well at Georgia Tech. The legislature has designated \$15 million to microelectronics to be matched by an equivalent amount from industry and external sources. I have been teaching one quarter and then doing research and writing the next quarter. Compared to A&M this is the "soft" life.

Very truly yours,

Phillip E. Allen
Schlumberger Professor
of Electrical Engineering

GEORGIA INSTITUTE OF TECHNOLOGY
School of Electrical Engineering
Atlanta, GA 30332

(404) 894-6251

March 18, 1985

MEMORANDUM

TO: Ralph K. Cavin, III, Program Manager, Design Sciences, SRC

FROM: Phillip E. Allen, Professor, School of Elect. Engr., Georgia Tech

SUBJECT: Quarterly Report, Dec. 1, 1984 to Feb. 28, 1985, SRC Project
84-07-051

INTRODUCTION

This Quarterly Report covers the period from December 1, 1984 to February 28, 1985, for the SRC sponsored research program at Georgia Institute of Technology titled "A Multilevel Analog Integrated Circuit CAD Program". The objective of this research program is to develop and implement a multilevel analog IC design program which will increase analog IC design productivity, insure testability, and be compatible with CAD techniques for digital IC design.

The program has made good strides in the second quarter of effort. The AIDE2 program has been debugged and applied in a classroom situation to simulate and layout switched capacitor filters. A technology independent database has been developed for 3-5 micron, single-well, CMOS technologies. As a result, the AIDE2 program is not restricted to one type of technology but can be applied to other technologies by changing the generic parameter set and selecting the appropriate geometric scaling constant. The testability feature of the program has been developed and simulated. Test circuits, manually implemented, will be fabricated. The blocks for the high-level design program for A/D converters have been designed and will be fabricated to obtain experimental verification of simulated performance. All software tools have been installed and are available on the VAX 11/780.

ACCOMPLISHMENTS

The primary objective of the research during this time has been to get the research tools in place and to debug and apply AIDE2. This has been successfully accomplished. The AIDE2 program is capable of simulating and laying out any circuit which can be constructed from the programmable linear block. The linear block consists of an op amp with the noninverting input grounded and various types of inputs to the inverting input. The types of inputs include switched and unswitched capacitors of any number or any size. The designer interacts with AIDE2 using the C programming language to describe his circuit. An example for a sixth order, low-pass, Chebyshev filter is given in Table 1. The filter is a cascade of three, second-order sections and has a cutoff frequency of 4 KHz, a passband ripple of 0.5 dB, and a clock frequency of 100 KHz. This table shows the manner in which the user can describe his circuit and to accomplish the simulation and layout. Fig. 1 shows the simulation results which are compared to SCANAL, an alternate switched-capacitor simulator. Fig. 2 shows the layout generated by AIDE2. This circuit and several others will be fabricated at the Harris Semiconductor Corp. in June 1985.

The technology independence of AIDE2 includes the definition of a generic technology, a generic parameter set, and a generic design rule set. All three of these aspects have been completed. The generic technology is illustrated in Table 2 where the CIF layers of the technology have been identified. It is seen that the technology is single well (n or p), has two polysilicon layers, and a mask for shielding polysilicon from implantation. Some of the layers have been defined in terms of their function rather than their composition in order to emphasize the generic aspects. For example, the top plates of the capacitors could be metal or upper polysilicon. The generic parameters are shown in Table 3 and include process defined parameters and user defined parameters. Each of the process defined parameters has a test vehicle which when fabricated for a given process will allow the designer to experimentally determine the values of the parameters. Table 4 shows the test structure and the method used to experimentally derive the process parameter. These parameters are used to obtain the model parameters for simulation. The generic design rules are illustrated in Table 5 and are in terms of lambda. The designer of a new block enters his design in units of 1/100 of lambda. The program chooses the value of lambda as half the minimum feature size for the particular technology.

A test methodology has been developed which can be employed by AIDE2 to physically isolate all or parts of a circuit for experimental testing after fabrication. The topology is determined by the contents of an on-chip serial register. A test cell has been designed which fits the AIDE2 layout format and allows the output/input ports of a cell to be connected in three modes. The first is the regular connection, the second is to an external test pad, and the third is to both the regular connection and external test pad. The test cell is in a slice configuration and can be expanded to the number of ports to be tested. The test cell is located next to the cell to be tested and the connections are made via the channel by the normal routing algorithm of AIDE2. An example of the capability of the test cell is shown in Fig. 3 for a third-order filter. There are 3 blocks in the filter with the input and output ports identified. The contents of 12 serial shift registers which determine the various test configurations is illustrated. This example is being manually implemented in the Harris MPC in order to examine the influence of the test circuit upon the filter performance.

A high level design program which uses AIDE2 to automatically design, simulate, and layout successive approximation type A/D converters is being developed. The topology for the successive approximation register has been selected. It uses a bit-slice approach allowing for programmability of the number of bits in the converter. Simulation techniques suitable for the converter are being examined since the present simulation capabilities of AIDE2 are not adequate for A/D converters. The various programmable blocks of the successive approximation converter are being designed and will be fabricated and tested to provide experimental data on accuracy, speed, and other performance parameters necessary to complete the high level design program.

All software tools have been installed on the VAX 11/780. Two AED workstations have been received. The bit-pads necessary to use these workstations have not yet been received. PSPICE, a version of SPICE that runs on the IBM PC has been received and installed. It works very well and will be a key aspect of our efforts to translate some of the analog CAD software to a PC environment.

A mini-design review of our SRC program was given to Mr. Robert Webb of Harris Semiconductor, Corp. on February 11, 1985 when he visited the Georgia Tech campus for discussions concerning the MPC interaction between Harris and Georgia Tech. A copy of the presentation was forwarded to SRC earlier.

PLANS

Plans for the third quarter generally involve bringing AIDE2 to the point where it can be used by others. This will include documentation, the development of a menu driven input routine so that the extensive amount of input shown in Table 1 for the 6th-order filter can be avoided, and the generation of designs to be fabricated on the Harris MPC in June. We would also like to be able to fabricate some of the same designs at Silicon Systems in order to evaluate the success of our technology independence efforts. New blocks which include a comparator, a sign multiplexer, an op amp with an output stage, the blocks for the successive approximation A/D converter, etc. will be added to the AIDE2 library. The simulation capability of AIDE2 will be generalized and extended to these new blocks.

Fig. 4 shows the plans and progress for this project. It is seen that we are on schedule except for the testing of the 184 MPC circuits and the development of the nonlinear block. The fabrication of the 184 MPC circuits were delayed until just recently and are not expected until April or May. The nonlinear block capability will be developed this quarter by the new blocks which will be added to the AIDE2 library. An important part of this research will be to identify a benchmark design that can be implemented by AIDE2 and compared to the performance of the original circuit.

Our plans for the project are to have a design review sometime early in the summer of 1985. At this review, we would like to demonstrate the AIDE2 program and to have the software and documentation ready to release to any SRC companies which may be interested in applying our research results. This release will be that of AIDE2.0 which will not include the testability, the multilevel simulation, or high level programs for analog circuit design. We consider this objective to very important to begin transferring the research to industry and to gain feedback from industry users.

On May 1985, a paper titled "AIDE2: An Automated Analog IC Design System" will be presented at the IEEE Custom Integrated Circuits Conference at Portland, OR. A copy of this paper is appended to this report. An MS thesis by Steve Bily at Texas A&M University was completed and a copy will be sent to SRC as soon as available. This thesis describes the design and development of the prototype AIDE and the experimental results. Other publications are planned for the 1985 Midwest Symposium and will be included in the next report.

PERSONNEL

The personnel involved with the project are Phillip Allen (Principal Investigator and Professor), Pat Barton (MS student), Raouf Halim (MS student), Seong Hong (PhD student), Kwang Yoon (PhD student), and Bruce Young (MS student). Mrs. Sherri Brenner has been supported part-time as a secretary. Mrs. Brenner, Mr. Halim, and Mr. Yoon have been supported by SRC during this quarter. Mr. Romeo Asibal (MS-PhD student) will be hired starting next quarter using SRC funds. Mr. Asibal will take over Mr. Halim's responsibilities when he graduates in June 1985.

EXPENDITURES

The expenditures for the period of December 1, 1984 through February 28, 1985 are as follows:

	<u>Dec. 1984</u>	<u>Jan. 1985</u>	<u>Feb. 1985</u>	<u>Total</u>
Personal Services	2612.21	2748.02	8621.31	13,981.54
Fringe Benefits	117.23	83.30	139.64	340.17
Materials & Supplies	544.27	110.35	215.30	869.92
Computer	-	-	-	-
Travel	1071.60	-	-	1071.60
Overhead	2402.96	1626.74	1537.11	5566.81
Totals	<u>6748.27</u>	<u>4568.11</u>	<u>10513.36</u>	<u>21830.04</u>


```

#include <stdio.h>
#include <math.h>
#include "/usr/src/aide2/lib/all.ext"

main () /* 6th order Chebychev low pass filter */
{
/*
Filter circuit description driver program
*/
Begin7("lp6fil",n1,n2,n3,n4,n5,n6,n7)
{
int out_net,ninputs,in_nets[10],in_phase[10],t_phase[10];
int plotype = 0;
float fc,cap[10];
char spice_file[20],cif_file[20];
printf("\n SPICE file name : ");
scanf ("%s",spice_file);
fc=100e3;
begin_ckt(spice_file,"SIXTH ORDER LOW PASS FILTER",fc);

/* BLOCK #1: */
out_net=n2; ninputs=4 ;
in_nets[1]=n1 ; in_nets[2]=n2 ; in_nets[3]=n2 ; in_nets[4]=n3 ;
in_phase[1]=2; in_phase[2]=3; in_phase[3]=2; in_phase[4]=2;
t_phase[1]=2; t_phase[2]=3; t_phase[3]=2; t_phase[4]=2;
cap[1]=6. 5128; cap[2]=25. 62; cap[3]=1; cap[4]=6. 5128;

linear_block("LIN1",spice_file,out_net,ninputs,in_nets,in_phase,
t_phase,cap,fc);
/* BLOCK #2: */
out_net=n3; ninputs=2;
in_nets[1]=n2; in_nets[2]=n3;
in_phase[1]=2; in_phase[2]=3;
t_phase[1]=1; t_phase[2]=3;
cap[1]=6. 5128; cap[2]=25. 62;

linear_block("LIN2",spice_file,out_net,ninputs,in_nets,
in_phase,t_phase,cap,fc);

/* BLOCK #3: */
out_net=n4; ninputs=4;
in_nets[1]=n3; in_nets[2]=n4; in_nets[3]=n4; in_nets[4]=n5;
in_phase[1]=2; in_phase[2]=3; in_phase[3]=2; in_phase[4]=2;
t_phase[1]=2; t_phase[2]=3; t_phase[3]=2; t_phase[4]=2;
cap[1]=1. 8103711; cap[2]=2. 3568809;
cap[3]=1; cap[4]=1. 8103711;

linear_block("LIN3",spice_file,out_net,ninputs,in_nets,
in_phase,t_phase,cap,fc);

```

(Continued on next page)

Table 1 - Example of C-Language description of a 6th-order filter for the AIDE2 program.


```

/* BLOCK #4: */
    out_net=n5; ninputs=2;
    in_nets[1]=n4; in_nets[2]=n5;
    in_phase[1]=2; in_phase[2]=3;
    t_phase[1]=1; t_phase[2]=3;
    cap[1]=1.8103711; cap[2]=2.3568809;

    linear_block("LIN4", spice_file, out_net, ninputs, in_nets,
    in_phase, t_phase, cap, fc);

/* BLOCK #5: */
    out_net=n6 ; ninputs=4;
    in_nets[1]=n5; in_nets[2]=n6; in_nets[3]=n6; in_nets[4]=n7;
    in_phase[1]=2; in_phase[2]=3; in_phase[3]=2; in_phase[4]=2;
    t_phase[1]=2; t_phase[2]=3; t_phase[3]=2; t_phase[4]=2;
    cap[1]=1; cap[2]=10.0417; cap[3]=1.46275; cap[4]=1;

    linear_block("LIN5", spice_file, out_net, ninputs, in_nets,
    in_phase, t_phase, cap, fc);

/* BLOCK #6: */
    out_net=n7 ; ninputs=2 ;
    in_nets[1]=n6; in_nets[2]=n7;
    in_phase[1]=2; in_phase[2]=3;
    t_phase[1]=1; t_phase[2]=3;
    cap[1]=1; cap[2]=10.0417;

    linear_block ("LIN6", spice_file, out_net, ninputs, in_nets,
    in_phase, t_phase, cap, fc);

    interconnect(spice_file);
    input_source(spice_file, "IN", n1, 2, 0.0, 1.0);
    anastat(spice_file);
    output(spice_file, "AC", n7, 1);
    end_ckt(spice_file);
    Standard1("pad", "pad1", n1); /* input pad */
    Standard1("pad", "pad2", n7); /* output pad */
    Standard1("pad", "pad3", n3); /*first stage out */
    Standard1("pad", "pad4", n5); /*second stage out */

    Analog_layout(100);
    printf("\n CIF file name: ");
    scanf("%s", cif_file);
    printf("\n enter 1 for a bounding box plot");
    printf("\n      0 for a detailed plot : ");
    scanf("%d", &plotype);
    CIF_out(cif_file, plotype);

}
End();

```

Table 1 (Continued) - Example of a C-Language description of a 6th-order filter for the AIDE2 program.

1. CIF layers and color-code for X-well process.

Layer	Code	Color
x-well	CXW	brown
x-moat	CXM	orange
xb*-moat	CXBM	green
x-gate	CXG	red
xb-gate	CXBG	red
capacitor bottom plate	CCBP	red
capacitor top plate	CCTP	purple
resistor	CR	red
metal	CM	blue
contact	CC	black
pad-opening	CPO	black
mask	CMS	black
x+ guard ring	CXGR	sky

* If X is p(or n) type, Xb may be n(or p) type.

Table 2 - Generic technology definition.

PROCESS DEFINED PARAMETERS

Interlayer capacitance
Doping levels
Junction depth
Lateral diffusion length
Oxide charge density
Sheet resistance

USER DEFINED PARAMETERS

Area of the source and drain
Periphery of the source and drain
Temperature
Channel length
Channel width
Minimum physical resolution unit (Λ)

Table 3 - Generic parameters and their categorization.

TEST STRUCTURE	TEST METHOD	PARAMETER DETERMINED
Poly-gate MOS Capacitor	C-V measurement at high frequency	Oxide Capacitance Oxide charge density Substrate doping level
Poly-gate to Well MOS Capacitor	C-V measurement at high frequency	Well doping level
Reverse bias junction capacitor	Reverse bias capacitance measure- ment	Source and Drain Doping level
P MOSFET	P MOSFET channel conductance	Hole surface mobility
N MOSFET	N MOSFET channel conductance	Electron surface mobility
Cross-bridge Resistor	Four point probing	Sheet resistance
	Combination of C-V, Channel conductance, and Four point probing Measurement	Junction depth

Table 4 - Test structure and the test method to experimentally determine the generic parameters.

Design rules for x-well process.

Layer	Minimum Dimension (lambda)
1. X-well	
1A. width	4
1B. spacing	7
2. Moat	
2A. width	2
2B. spacing to x-well	
2B1. x-moat contained in x-well	2
2B2. x-moat not touching x-well	5
2B3. xb-moat contained in x-well	3
2B4. xb-moat not touching x-well	5
2C. spacings to other moat (inside or outside well)	
2C1. x-moat to x-moat	3
2C2. xb-moat to x-moat	3
2C3. xb-moat to xb-moat	3
2D. spacing to resistor	2
2E. source-drain overhang of x-gate (or xb)	3
2F. x+ guard ring structure	
2F1. extends outside x-well	3
2F2. extends inside x-well	1
3. X+ Guard Ring	
3A. width	4
3B. spacing	2
3C. spacing to x-moat (or xb) outside x-well	3
3D. spacing to x-moat (or xb) inside x-well	3
4. X-Gate	
4A. width	2
4B. spacing	2
4C. spacing to Moat	
4C1. x-moat (x-gate in x-moat)	2
4C2. x-moat (x-gate in field)	2
4C3. xb-moat (x-gate in xb-moat)	2
4C4. xb-moat (x-gate in field)	2
4D. overhang of x-gate	2
5. Xb-Gate	
5A. width	2
5B. spacing	2

(Continued on next page)

Table 5 - Design rules for the generic parameter set.

5C. spacing to Moat	
5C1. x-moat(xb-gate in x-moat)	2
5C2. x-moat(xb-gate in field)	2
5C3. xb-moat(xb-gate in xb-moat)	2
5C5. xb-moat(xb-gate in field)	2
5D. overhang of xb-gate	2
6. Capacitor bottom plate	
6A. width	2
6B. spacing	2
6C. spacing to x-moat(or xb)	2
6D. overhang of capacitor top plate	2
7. Capacitor top plate	
7A. width	2
7B. spacing	2
7C. spacing to x-moat(or xb)	2
8. Resistor	
8A. width	2
8B. spacing	2
8C. spacing to x-moat(or xb)	2
8D. spacing to x-gate(or xb)	2
9. Metal	
9A. width	3
9B. spacing	3
9C. Bonding Pad	
9C1. size	44 X 44
9C2. spacing(pad metal to pad metal)	20
9C3. spacing to x-moat	12
9C4. spacing to xb-moat	12
9C5. spacing to metal circuitry	12
9C6. spacing to x-gate(or xb)	12
9C7. spacing to resistor	12
9C8. spacing to capacitor bottom plate	12
9D. Probing Pad	
9D1. size	30 X 30
9D2. spacing(pad metal to pad metal)	12
9D3. spacing to Bonding Pad	12
9D4. spacing to x-moat	12
9D5. spacing to xb-moat	12
9D6. spacing to metal circuitry	12
9D7. spacing to x-gate(or xb)	12
9D8. spacing to resistor	12
9D9. spacing to capacitor bottom plate	12

(Continued on next page)

Table 5 - Continued.

10. Contacts

10A. size	2 X 2
10B. spacing	2
10C. spacing to x-gate(or xb)	2
10D. spacing to x-moat(or xb)	2
10E. metal overlap of contact	1
10F. x-moat(or xb) overlap of contact	1
10G. resistor overlap of contact	2
10H. x-gate(or xb) overlap of contact	2
10I. capacitor bottom plate overlap of contact	2
10J. capacitor top plate overlap of contact	2

11. Pad opening

11A. Bonding Pad opening	40 X 40
11B. Probe Pad opening	26 X 26

12. Mask

12A. spacing to x-gate(or xb)	2
12B. overhang of resistor	3

Table 5 - Continued.

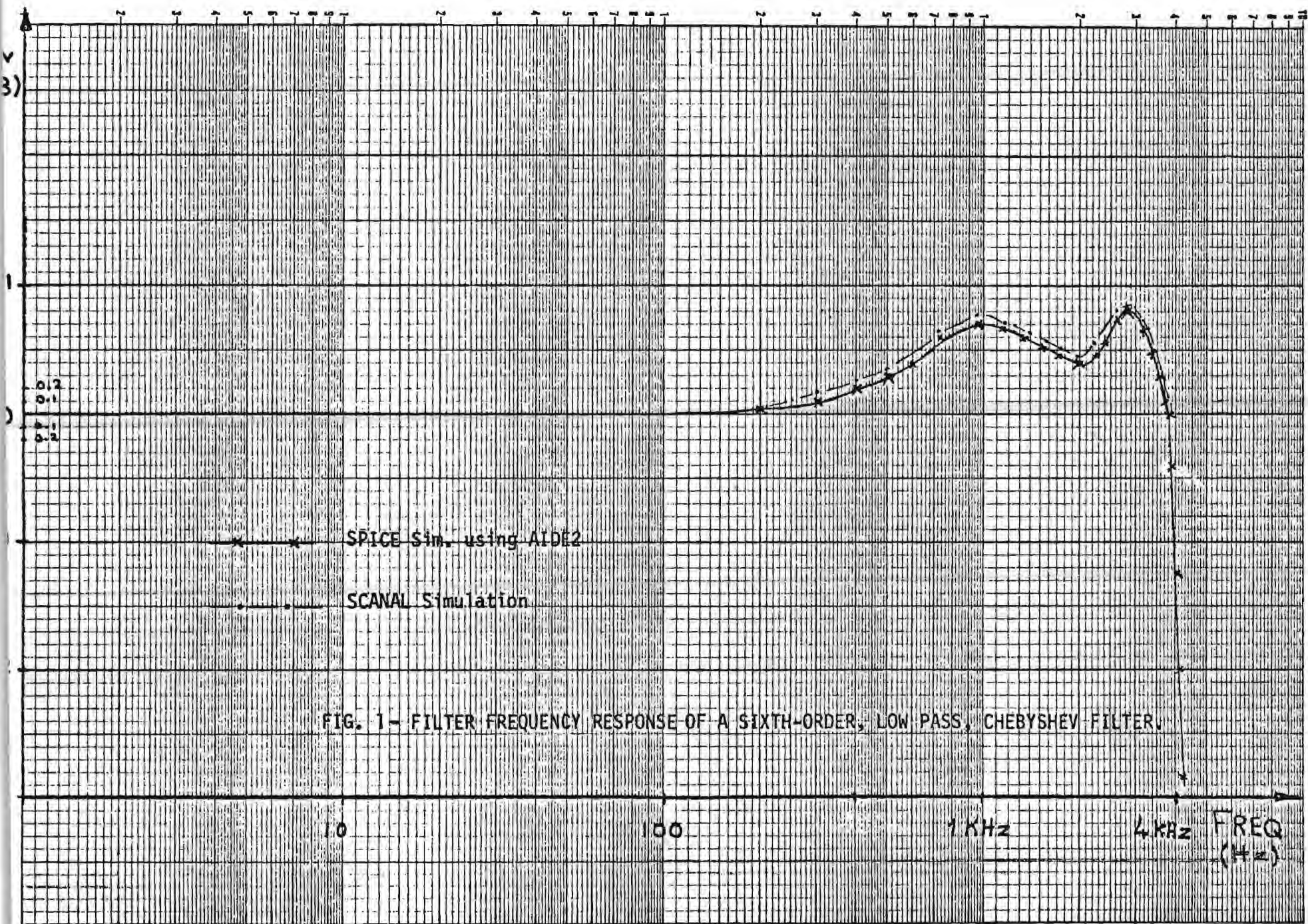


FIG. 1- FILTER FREQUENCY RESPONSE OF A SIXTH-ORDER, LOW PASS, CHEBYSHEV FILTER.

cifplot* Window: 0 162000 0 173000 --- Scale: 1 micron is 0.0048 inches (122x)

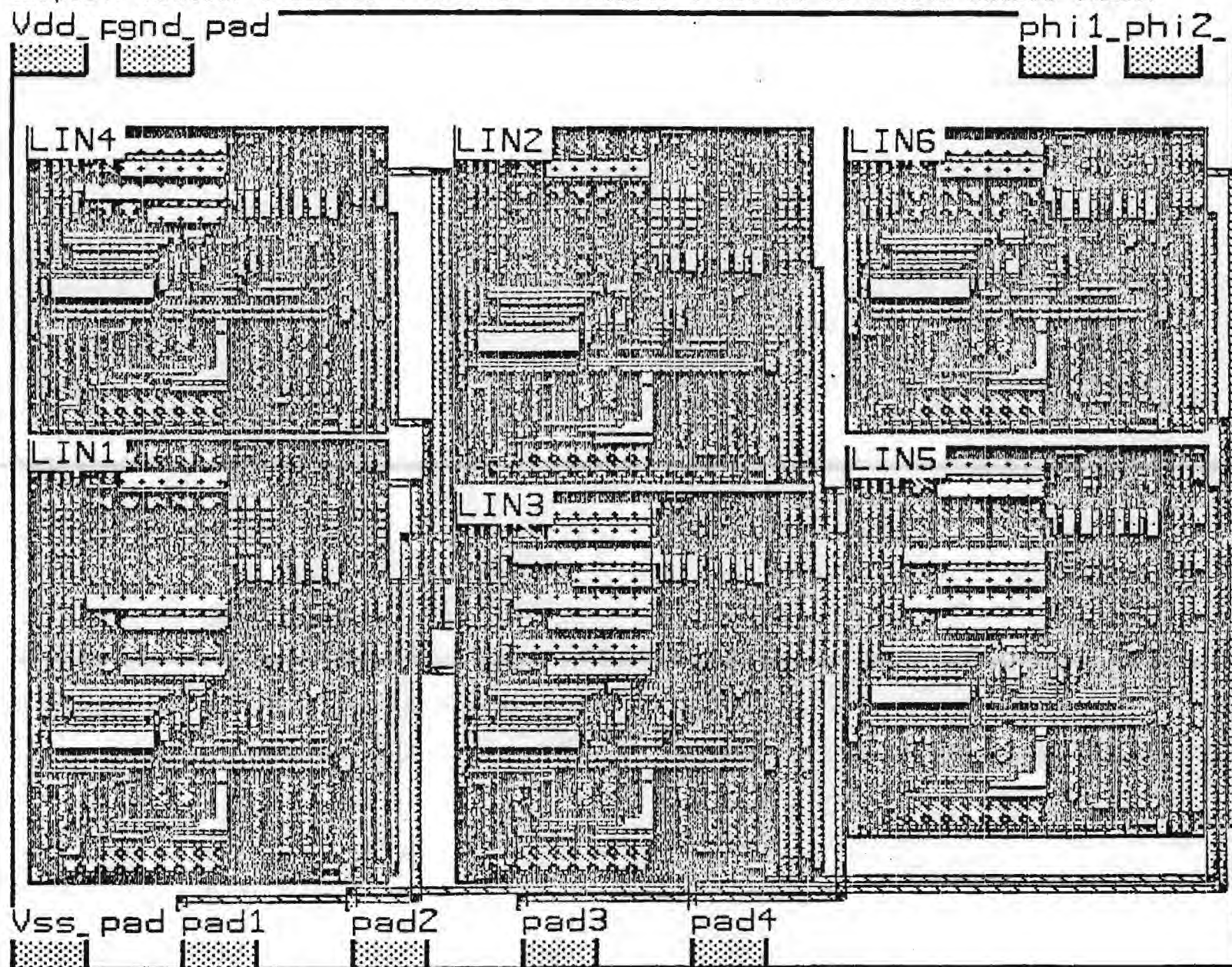


Fig. 2 - AIDE2 layout of the sixth-order, low-pass, Chebyshev filter.

		BLOCK 1					BLOCK 2			BLOCK 3							
	PORT		11	21	31	41	61	12	22	62	13	23	33	63	INPUT	OUTPUT	
	NORMAL		0	0	0	0	0	0	0	0	0	0	0	0	21	63	
T E S T	SYSTEM		0	1	0	0	0	0	0	0	0	0	0	1	21	63	
	SUBSYSTEM		1	1	0	0	0	0	1	1	0	0	0	0	11,21,22	62	
	BLOCK	1	1	1	1	1	1	0	0	0	0	0	0	0	11,21,31,41	61	
		2	0	0	0	0	0	0	1	1	1	0	0	0	0	12,22	62
		3	0	0	0	0	0	0	0	0	0	1	1	1	1	13,23,33	63

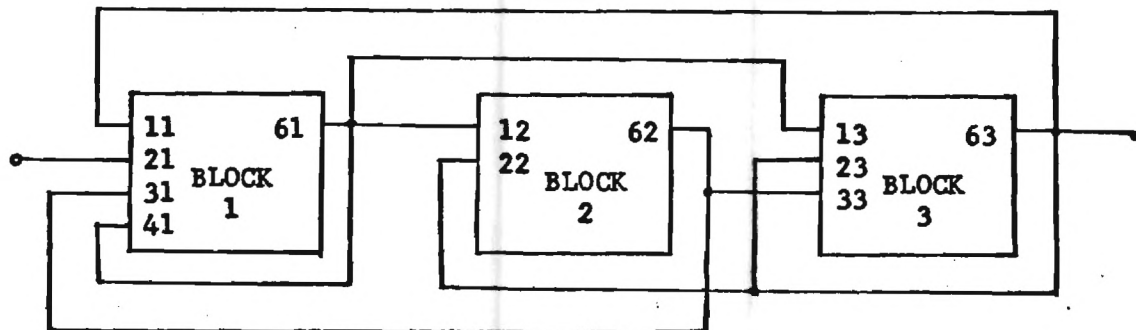


Fig. 3 - Third-order filter block diagram and the values of the 12 serial shift registers which will achieve various test isolations and connections.

PLANS FOR SRC RESEARCH IN ANALOG CAD

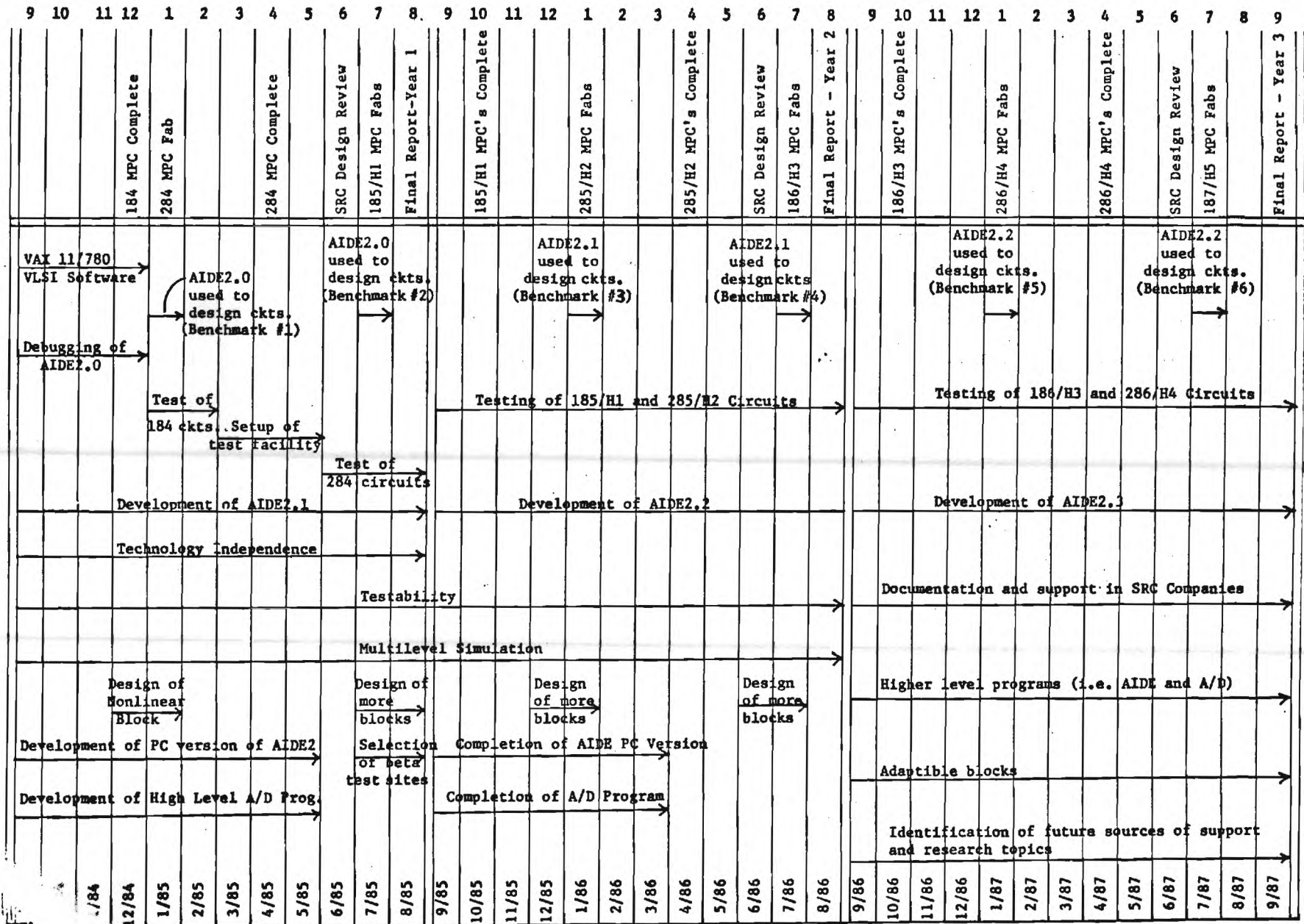


Fig. 4 - Plans for SRC research program in analog CAD.

AIDE2: AN AUTOMATED ANALOG IC DESIGN SYSTEM

Phillip E. Allen

School of Electrical Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Edmond R. Macaluso *
Stephen F. Bily
Ashok Nedungadi

Dept. of Electrical Engineering
Texas A&M University
College Station, TX 77843

ABSTRACT

A technique for the automated design of analog integrated circuits is presented. This technique employs analog module compilers to produce the cells in an analog standard cell layout system. The design of a CMOS switched capacitor module compiler is described along with the technique used for developing "soft macros" in the C programming language.

INTRODUCTION

The objective of this paper is to present a technique for the automated design of analog integrated circuits. This technique provides automated design of various types of analog circuits from a library of parameterized cells. A second generation design system called Analog Integrated circuit Design (AIDE2) has been developed and used in the design of switched capacitor circuits.

Automation is provided for the component layout, circuit synthesis and system layout phases of the design process. Three concepts are employed in AIDE2:

1. Module compilers are used to generate the primitive components. This is needed since analog components may require a wide variety of devices such as capacitors or transistors with different sizes.
2. Soft macros are specified in the C programming language. The term "soft macro" or "functional subnetwork" is familiar in semicustom design and simply refers to an interconnection of cells [1]. The soft macro technique used in AIDE2 has many capabilities and yet is simple to implement and use.
3. Automatic placement and routing is used for system layout. This is done with analog standard cell techniques.

The goal of this research is to reduce the design time and complexity in analog integrated circuit design, especially in the case of analog and digital circuits on the same substrate. Some of the other benefits of automated design over manual design are that fewer errors are introduced, systematic techniques for shielding and testing may be used; and chip designer leverage is increased especially in the case of module compilers. The trade off with automated design is the area efficiency and the optimization of component design to the circuit function.

The first part of the paper reviews results of the prototype AIDE system which was developed to test the feasibility of designing analog circuits with a general purpose block. Some encouraging results were obtained with this subsystem. The design of a

switched capacitor module generator used in AIDE2 is presented next. This cell overcame many of the deficiencies in the original AIDE cell; and the design of the cell helped to debug new CAD tools provided in AIDE2 for the development of module compilers. The paper concludes with an example of how to build soft macros and how to describe circuits in AIDE2.

PROTOTYPE SYSTEM

The organization of the prototype AIDE system and the various applications of the system are discussed elsewhere [2], [3], [4]. A third-order, elliptic, low-pass filter is shown in Figure 1.

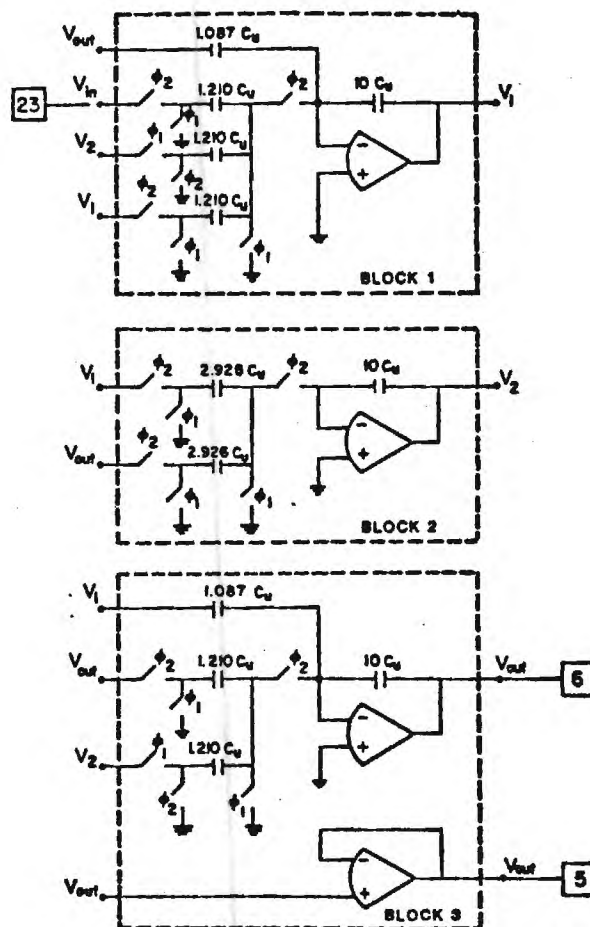


Figure 1 - A third-order, elliptic, low-pass filter

* on leave from VR Systems/ A Tektronix Co.

This figure illustrates the type of circuit in which the layout is done automatically by the AIDE system. The dotted lines represent three separate blocks which are interconnected automatically as standard cells. Each block is programmed with the analog module compiler in AIDE. The experimental results, shown in Figure 2, correspond well with the simulated results. The dashed line is the SPICE simulation of the passive LC ladder, the solid line is the SWITCAP simulation of the SC filter, and the dotted line is the experimental result from the fabricated circuit [3].

The AIDE module compiler has several limitations which limit the type of circuits which could be implemented. First, the block has a maximum of four inputs in which unused inputs result in wasted area. The largest-to-smallest ratio of capacitors in each block is 10. NMOS switches are used rather than complementary switches which results in a limited signal swing. In addition, the prototype AIDE system was not designed with a cell database or CAD tools for developing new cells. Aside from these problems, a wide variety of circuits have been designed with the AIDE system in both a laboratory and classroom environment.

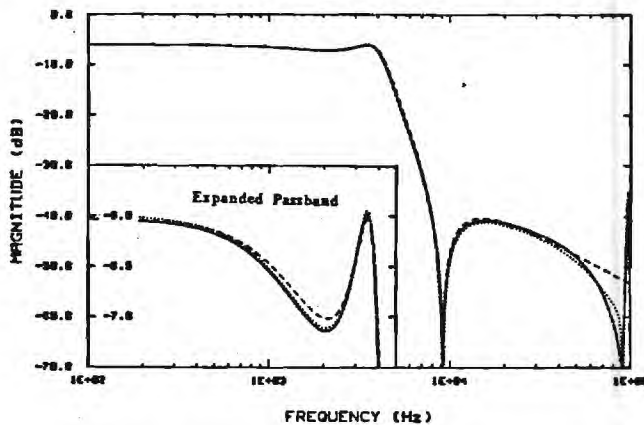


Figure 2 - Simulated and experimental frequency response of the circuit shown in Figure-1

AIDE2

From the experience gained in the development of the prototype AIDE system, several requirements were outlined and implemented in AIDE2:

1. The SC module compiler was redesigned to overcome the limitations in the prototype AIDE system.
2. CAD tools were developed to aid in the development of various module compilers.
3. A circuit description technique was developed to facilitate development and integration of synthesis programs.
4. The standard cell system was rewritten to provide efficient layout and to not restrict the number or types of cells.

The details of the AIDE2 system can be found elsewhere [5]. The remainder of this paper presents the design of the switched capacitor module compiler in AIDE2 and illustrates how circuits are described with the AIDE2 procedural circuit description language.

An SC Module Compiler

The generic schematic of the AIDE2 switched capacitor module compiler is illustrated in Figure 3. The parameters for this cell are the number of inputs (1-unlimited), the capacitance value for each input (0-unlimited) and the clock phasing for each of the input and transfer switches. The switch configurations are parasitic insensitive [6] in which the input switches are complementary NMOS/PMOS to allow full swing of the input signals. Any of the inputs may be configured in a parallel, series, direct capacitor or direct op amp connection. There are practical limits to the number of inputs or capacitance at each input, however, these limits have yet to be investigated. An 8th-order filter, to be described later, has one cell with a capacitance ratio of 90 to 1.

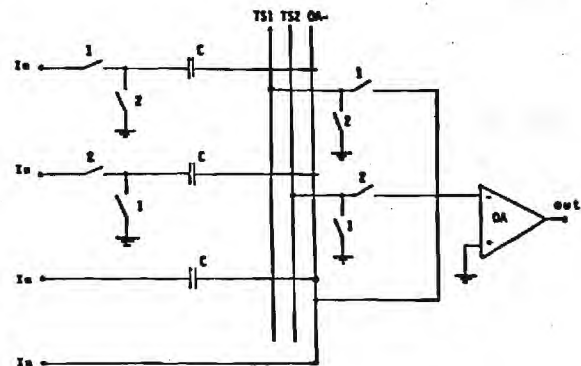


Figure 3 - Generic schematic of the SC module compiler

A programmed SC block is illustrated in Figure 4. This particular block is programmed with five inputs to illustrate the various input configurations. The op amp, which is not illustrated, makes connection by abutment on the right side of the cell.

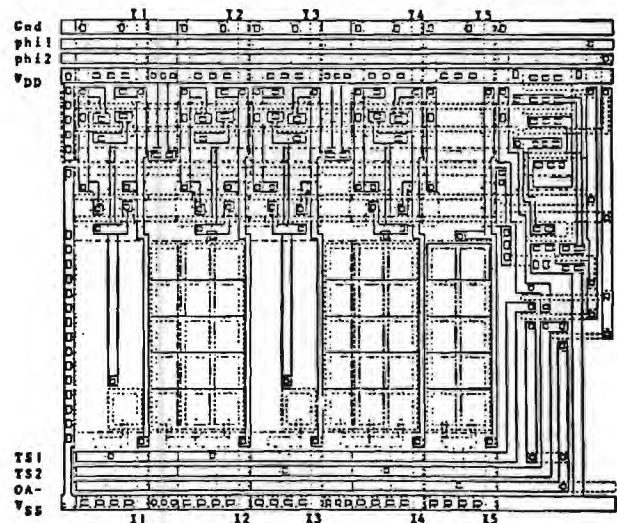


Figure 4 - A programmed SC block with five inputs

The first input, labeled I1, is on the far left of the cell and its port is equipotential on both the top and bottom. The capacitance of I1 is one rectangular unit, approximately 0.3 pF, which is located on the lower half of the input. The input switch section for this input is located in the upper half of the I1 input slice.

As illustrated in the generic schematic, the transfer switches are shared by all inputs and connection to a transfer switch is made by placing a contact cut on the appropriate transfer switch buss (TS1, TS2 or OA-). Input I1 of the programmed block makes connection to the TS1 buss. The transfer switch busses run on the lower portion of the cell to the transfer switch slice on the far right.

The other inputs are programmed similarly and any additional inputs simply increase the width of the cell while maintaining constant height. Inputs I2 and I4, which have 15 capacitance units, illustrate how the additional capacitance is added to the inputs by simply increasing the width of the input slice.

On the top of the cell and in the switch sections, ground shielding is used to isolate the clock signals from the analog signals. This is done by placing poly-2 ground shields between metal and poly-1 intersections of the clock signals and analog signals. Shielding is also used wherever the clock busses cross channels. This technique provides complete segregation of the clock and analog signals on the entire chip.

The AIDE2 parameter preprocessor was used in the development of the software for the SC module compiler. Other tools were investigated including the LAP system [7] and tile packing [8]; however, these tools did not meet all of the requirements for analog compilation. These requirements included generation of both geometric and simulation data, scaling and repetition of data, low level flexibility and ease of use.

The procedure for designing a module compiler with the AIDE2 parameter preprocessor is as follows:

1. Design the generic schematic.
2. Floorplan the geometric layout.
3. Identify the fixed and programmable sections of the floorplan.
4. Layout the fixed sections with a graphical layout tool such as KIC2 [9] or Caesar [10] which produce a CIF file.
5. Hand code the programmable sections of the CIF layout file with the operators provided by the parameter preprocessor.
6. Hand code the software for the module compiler using the C language and functions provided by the parameter preprocessor.

Since the procedure requires hand coding of the software and a portion of the data file, the parameter preprocessor might not be considered easy to use; however, by the time the hand coding process starts, the circuit layout is well defined and the hand coding is routine. In addition, the programmable sections of a layout file typically constitute a small portion of the file. For example, the programmable section of the SC module compiler layout file was only 33 lines, about 4% of the file.

The parameter preprocessor provides a set of C language functions which correspond to a set of operators in the geometric file. Since the operators perform basic functions on data, they may also be used in simulation files if there are no syntax conflicts.

The C language function call for the SC module compiler is:

`liblk(obj-name, outnet, ninputs, innets, inphase, tphase, cap)`

in which all arguments are inputs. The arguments represent respectively the instance name for the object, the net the output connects to, the number of inputs, and arrays for the input net, switch phasing and capacitance of each input. Note how each argument is associated with the generic schematic in Figure 3.

The development of a module compiler is not a simple task and requires an experienced circuit designer with a basic understanding of programming. Once a design is completed, however, many different configurations may be generated either by system designers or synthesis programs and an intimate knowledge of analog integrated circuit design is not required.

Procedural Circuit Description

Circuit descriptions for AIDE2 are written in the C programming language. This technique allows great flexibility in the development of circuit design systems based on AIDE2. A circuit description as a C language function may represent a soft macro, a synthesis routine or even an interactive design system. Circuit descriptions are organized hierarchically in which the lowest level calls are to fixed geometry standard cells or module compilers. The netlist information and other characteristics are passed as function arguments. The AIDE2 organization is illustrated in Figure 5.

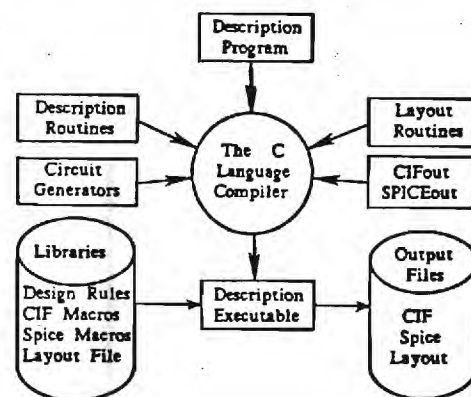


Figure 5.- Organization of AIDE2

To illustrate this concept, a soft macro was created for the biquad circuit topology shown in Figure 6. This soft macro is a C language routine with the following function call:

`BiquadSC1(object-name, input-net, output-net, cap-array)`

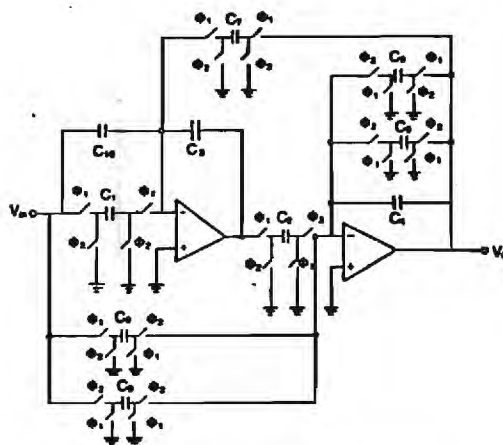


Figure 6 - A general biquadratic SC filter

The BiquadSC1() routine calls the SC module compiler twice. The first call specifies an SC linear block with four inputs and the second call specifies a block with six inputs. All of the topology of the biquadratic circuit is stored in BiquadSC1. There are three nets in this circuit in which the connections to the input and output nets are passed as arguments and a local net is created internally. The 10 capacitor values are passed into the routine in cap-array.

EXAMPLE

As an example, an 8th-order Chebyshev bandpass filter is considered. The specifications for the filter are its Bode plot characteristics which are entered into the AROMA filter design program [11]. This program calculates the capacitance values for the four cascaded biquad sections using the general SC biquad filter shown in Figure 6.

The AIDE2 circuit description for the filter is described in the C programming language as follows:

```
#include "aide2-macros"
float cap4[11] = {0,0,91,56,60,2,0,15,2,1,1}
float cap3[11] = {0,0,78,48,52,2,0,18,2,1,1}
float cap2[11] = {0,0,36,22,23,2,0,6,2,1,1}
float cap1[11] = {0,0,34,21,22,2,0,7,2,1,1}
main() { /* 8th order SC filter */
  Begin5("bandpass", n1, n2, n3, n4, n5) {
    BiquadSC1("Block4", n4, n5, cap4);
    BiquadSC1("Block3", n3, n4, cap3);
    BiquadSC1("Block2", n2, n3, cap2);
    BiquadSC1("Block1", n1, n2, cap1);
    Standard1("pad", "output_pad", n5);
    Standard1("pad", "input_pad", n1);
  }
  End() Analog-layout(100); CIF-out("F8.cif",1); }
```

The Begin5 statement is an AIDE2 macro which expands into function calls that define the five nets, n1 to n5. The four biquads are called and pads are connected to the input and output nets. Note that the StandardX() routine is used for all fixed geometry cells in which X is the number of i/o's for the cell. The Analog-layout() routine invokes the analog standard cell layout system with an aspect ratio of 100:100. The resulting layout, shown in Figure 7, is 90x100 mils using a 5 micron CMOS process. For clarity, the details of the blocks and the system busses are not shown. The details of each block are similar to that shown in Figure 4 except that the capacitor values in the example are very large. The total capacitance for all four biquads is 610 units. Fabrication results are pending a conversion to a new process.

CONCLUSIONS

A technique for the automated design of analog integrated circuits has been presented. This technique borrows the concepts of module compilation and standard cell layout from digital design automation and combines them to produce a practical analog design system. Since the height of analog standard cells is typically five times that of digital counterparts and the functional requirements are more complicated, the technique described leads to efficient analog designs whereas the application to digital design is more limited.

Areas of future research in analog IC design include the development of new module compilers and synthesis routines and new techniques in the areas of design for testability and multilevel simulation of analog circuits. Further developments in IC CAD will also be pursued including expansion of the AIDE2 standard cell capabilities, technology independent considerations and the incorporation of a more versatile building block layout system.

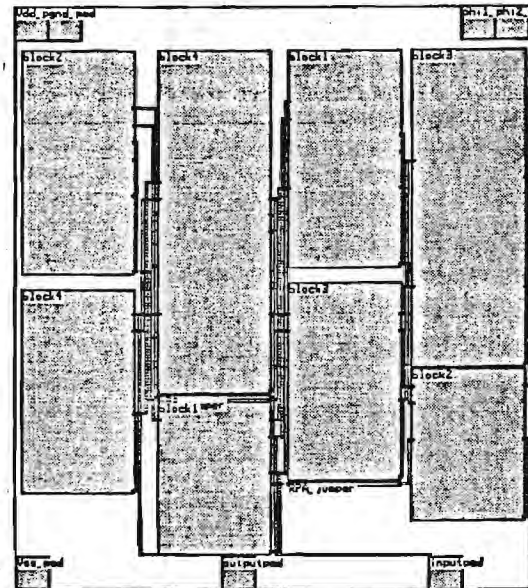


Figure 7 - AIDE2 layout of the 8th-order SC bandpass filter

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of this work by the Semiconductor Research Corporation and the Texas Engineering Experiment Station. Fabrication of the circuits was provided by Texas Instruments, Dallas. The op amp for the SC module compiler was designed by Horacio Nevarez and contributions to the implementation of the CAD tools were made by Mohsen Sharifin and David Burnett.

REFERENCES

- [1] A.J. Kessler and A. Ganesan, "Standard Cell VLSI Design: A Tutorial," *IEEE Circuits and Devices Magazine*, vol. 1, no. 1, pp. 17-33, Jan. 1985
- [2] P.E. Allen et al., "AIDE - Analog IC Design Program," *Proc. of 26th Midwest Sym. on CAS*, pp. 622-626, Aug. 1983
- [3] P.E. Allen et al., "The Use of a CAD Tool to Design Analog Integrated Circuits," *Proc. of ISCAS*, pp. 1223-1226, May 1984
- [4] S.F. Bily, "A Functional Cell Design Methodology for Analog Integrated Circuits," MS Thesis, Dept. of Electrical Engr., Texas A&M University, May 1985
- [5] E.R. Macaluso, "A Procedural Language for Automated Analog Circuit Design," MS Thesis, Dept. of Electrical Engr., Texas A&M University, Dec. 1984
- [6] P.E. Allen and E. Sanchez-Sinencio, *Switched Capacitor Circuits*, Van Nostrand Reinhold Co., New York, NY, 1984
- [7] S. Trimberger, J. Rowson, C.R. Lang and J.P. Gray, "A Structured Design Methodology and Associated Software Tools," *IEEE Trans. on CAS*, vol. 28, no. 7, pp. 619-633, July 1981
- [8] R.N. Mayo and J.K. Ousterhout, "Pictures with Parenthesis: Combining Graphics with Procedures in VLSI Layout Tools," *Proc. of the 20th DAC*, pp. 270-276, June 1983
- [9] G.C. Billingsley, Program Reference for KIC, Memo no. UCB/ERL M83/62, Electronics Research Laboratory, UC Berkeley, Oct. 83
- [10] J.K. Ousterhout, "Caesar: An Interactive Editor for VLSI Layouts," *VLSI Design*, vol. 2, no. 4, pp. 34-41, Oct. 1981
- [11] E. Sanchez-Sinencio, J.J. Lee and P.E. Allen, "An Area Optimized CAD Program for Cascade SC Filter Design," *ICCAD Digest of Technical Papers*, pp. 21-23, Nov. 1984

E-21-632



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

EPHONE: (404) 894-2961

July 29, 1985

Dr. Ralph K. Cavin, III
Program Manager, Design Sciences
Semiconductor Research Corporation
P.O. Box 12053
Research Triangle Park, NC 27709

SUBJECT: Contract No. 84-07-051, Project Director - Dr. P. E. Allen
Quarterly Report

Dear Dr. Cavin:

Enclosed please find the Quarterly Report for the period 3/1/85 - 6/30/85 for the above referenced project per contract specifications. If you have any questions, please contact Dr. Allen at 404/894-6251.

Sincerely,

Cindy Meyer
Admin. Asst.

CM

Enclosures

cc: OCA (2)



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

EPHONE: (404) 894- 6251

July 24, 1985

MEMORANDUM

TO: Ralph K. Cavin III, Director Design Sciences

FROM: Phillip E. Allen, Prof. School of Elect. Engr., Georgia Tech *Pea*

SUBJECT: Quarterly Report, March 1, 1985 to June 30, 1985, SRC Project No. 84-07-051

Introduction

This report covers the period from March 1, 1985 to June 30, 1985 for the SRC sponsored research program at Georgia Institute of Technology titled "A Multilevel Analog Integrated Circuit CAD Program". The objective of this research program is to develop and implement a multilevel analog IC design program which will increase analog IC design productivity, insure testability, and be compatible with CAD techniques for digital IC design.

Accomplishments

The primary efforts during this period have been the development of the AIDE2 CAD program and its debugging. Our major objectives have been to prepare for the SRC Design Review and Technology Transfer Seminar which were held at Georgia Tech on June 17 and 18, 1985 and to prepare AIDE2 designed circuits for the first multiproject chip at Harris Semiconductor to be fabricated in July 1985. We are presently working on two benchmark circuits to be fabricated. One is a fifth-order filter from Harris Semiconductor and the other a 14th-order filter from Silicon Systems Inc. The fifth-order filter is complete but the 14th-order filter requires some modifications in our block and may not be included in the July fabrication.

A Users Manual has been written for the AIDE2 program and tapes of the source code have been distributed to the 17 industrial members who attended the Technology Transfer Seminar on June 18, 1985. In addition, we repeated this seminar to two members from Rockwell International on July 22, 1985 who missed the June 18, 1985 Seminar. Several new blocks have been developed for AIDE2 and are being entered into the internal library. These blocks include a comparator and a clock multiplexer. The routing of the connections of the system to external pads has been completed. An algorithm for creating partial capacitors for the linear block has also been completed. One of the significant advances made has been to generalize the program to other fixed and programmable cells/blocks. This has required modification of the internal code and subroutines which utilize the geometric and simulation databases. The result is to provide an input to the designer at the low level of design.

Plans

Our plans include 1.) fabrication of AIDE2 circuits at Harris Semiconductor during late July, 2.) writing the Programmers Manual for AIDE2, 3.) Incorporating new blocks into the AIDE2 libraries, developing a program for automatically placing and routing test cells to achieve a specified experimental test capability, and 4.) completion of the 14th-order filter benchmark. We will also seek additional sources of fabrication to verify the ability of AIDE2 to design a given circuit using different technologies.

Personnel

The personnel involved with the project are Phillip Allen (Principle Investigator and Professor), Romeo Asibal (PhD student), Dessiree Ellsworth (MS student), Raouf Halim (MS student), Seong Hong (PhD student), and Kwang Yoon (PhD student). Mrs. Sherri Brenner has been supported part-time as a secretary. Mrs. Brenner, Mr. Asibal, Mrs. Ellsworth, Mr. Halim, and Mr. Yoon have been supported by SRC during all or part of this quarter.

Expenditures

The expenditures for the period of March 1, 1985 through June 30, 1985 are as follows:

	<u>March 1985</u>	<u>April 1985</u>	<u>May 1985</u>	<u>June 1985</u>	<u>Total</u>
Personal Services	10,131.00	5,689.00	5,689.00	11,684.00	33,139.00
Fringe Benefits	1,728.15	253.38	253.38	1,728.15	3,963.06
Materials & Supplies	2,026.62	5.54	99.93	458.60	2,590.69
Computer	-	-	-	-	-
Travel	-	-	400.00	-	400.00
Overhead	7,678.83	3,289.20	3,562.60	7,416.92	21,947.55
Totals	<u>21,564.60</u>	<u>9,237.12</u>	<u>10,004.91</u>	<u>20,829.08</u>	<u>61,635.70</u>

PRODUCED UNDER SRC CONTRACT? Yes/No/In Part SRC CONTRACT #: 84-07-051
 OTHER SOURCES OF SUPPORT? Schlumberger Chair Funds and GT E-Funds

NAME OF PERSON RESPONSIBLE

Phillip E. Allen
 Schlumberger Professor

NAME AND ADDRESS OF ORGANIZATION

School of Electrical Engineering
 Georgia Institute of Technology
 Atlanta, GA 30332

REPORT TITLE

Quarterly Report, July 1, 1985 to Nov. 30, 1985

REPORT AUTHOR(S)

Phillip E. Allen

TYPE OF REPORT

TIME COVERED
(if applicable)REPORT
DATEPAGE
COUNTDATE SUBMITTED
TO SRC

☒ Qtrly Progress
☐ Annual Report
☐ Final Report
☐ Thesis

FROM: 7/1/85
 TO : 11/30/85

2/20/86

4

☐ Pub. Preprint - submitted to... _____ date: _____
☐ Presentation - presented at.... _____ date: _____
☐ Other - please describe..... _____

ABSTRACT OF CONTENTS OF THIS SUBMISSION (not to exceed 1 single spaced page)

This report discusses the progress made on the development of a multi-level analog integrated CAD program. Accomplishments are described in the areas of AIDE2 CAD program development and documentation, testing and testability of circuits designed by AIDE2, development of a high level analog simulator, and the development of precision small signal models for sub-micron devices. Major accomplishments include the design of circuits by AIDE2 for fabrication, documentation of the AIDE2 program, and the layout of a chip for fabrication of submicron devices for testing and modelling.

SUBJECT KEYWORDS (circle keywords supplied on reverse - note any additions)
 LIST A: Analog, CAD, CMOS LIST B: IC/CAD Tools, auto. layout, dev. modeling

*** FOR SRC USE ONLY ***

DATE RECEIVED: _____

SRC PUBLICATION ID: _____



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

PHONE: (404) 894- 6251

February 20, 1986

MEMORANDUM

TO: Ralph K. Cavin, III, Director Design Sciences, SRC

FROM: Phillip E. Allen, School of EE, Georgia Tech

SUBJECT: Quarterly Report, July 1, 1985 to Nov. 30, 1985, SRC Project No.
84-07-051

Introduction

This report covers the period from July 1, 1985 to November, 30 1985 for the SRC sponsored research program at Georgia Institute of Technology titled "A Multilevel Analog Integrated Circuit CAD Program". This period of time really covers two consecutive quarters of research activity. The objective of this research is to continue the development of the AIDE2 CAD tool for analog integrated circuits and to incorporate a multilevel simulation and testing capability. A second objective of this research is to develop a precise, but computer efficient, small signal model for submicron MOS devices suitable for analog integrated circuit design.

Accomplishments

The research efforts have been divided into four major activities. These activities are (1) the application, debugging, development, and documentation of the AIDE2 CAD program, (2) testing and testability of circuits designed by AIDE2, (3) development of a high level analog simulator, and (4) the development of a precision small signal model for submicron MOS devices. The status and accomplishments of each of these activities will be individually described.

The AIDE2 CAD program reached its first level of maturity during the first part of summer 1985, when an SRC Technology Transfer Course was held at Georgia Tech to describe the AIDE2 program to SRC members. As with all large CAD programs, AIDE2 lacked in documentation and an extensive debugging that comes with usage. During the period of this report, efforts have been made to accomplish these goals. The Users Manual has been updated and a Programmers Manual is under development. The Programmers Manual will provide the information necessary to understand the development of the AIDE2 program allowing modification and intelligent usage. The AIDE2 program has been used to design approximately 17 switched capacitor filters ranging from simple to complex filters. Fig. 1 shows one of the quadrants of the multiproject chip (MPC) which is being fabricated at Harris Semiconductor that was designed entirely by the AIDE2 program. Considerable manpower was required to solve interface, design rule, and layout problems. Hopefully, the knowledge gained in evaluating these circuits will justify this effort. Also included on the

MPC are test circuits for evaluating the performance of a circuit that has been automatically laid out. Block performance concerns include, accuracy of gain and time constants, clock effects, offsets, frequency limitations, etc.

The second activity includes the incorporation into the AIDE2 program the ability to test analog integrated circuits. The objective is to be able to allow the AIDE2 user to experimentally replicate any simulation performed on an AIDE2-designed integrated circuit. The first problem that was encountered was that AIDE2 was incapable of simulating only a portion of the designed circuit. This problem was investigated and the best solution appears to be the use of a voltage amplifier at each port of each block. If the port is to be disconnected from the circuit the gain of the block is zero, otherwise the gain is unity. This should permit the simulation of any part of a circuit. Part or all of that subcircuit can be disconnected from the remainder of the circuit.

The third activity involves the development of a multilevel simulator capability for analog integrated circuits. It is desired to be able to simultaneously simulate an analog circuit at various levels. A linear graph theory approach is being investigated as a possible high level simulator. The user will enter a graph of the network by describing the edges of the graph. The format for the edge of a graph is a higher-order, rational polynomial. This permits the high level simulator to work with single elements such as a capacitor or complex functions such as an entire filter. The desired transmissions of the graph will be found using Mason's gain rule for calculating the transfer function between two nodes of a graph. A program has been developed for efficiently making the required calculations and has been applied to find the denominator of the transfer function. It is being extended to the numerator at the present time.

The last activity is the development of small signal models for submicron MOS devices for analog integrated circuit design. Test structures have been designed for a 1 micron process which will be fabricated by Texas Instruments. Devices lengths vary from 1, 2, 3, 5, and 10 microns for widths of 2, 3, 5, and 10 microns for both n-channel and p-channel devices. Also designed are test structures which can experimentally be used to measure the small signal model parameters of g_m , g_{ds} , g_{mbs} , C_{gd} , C_{db} , and C_{sb} . In addition, the software programs have been installed to automatically extract the BSIM model parameters using the HP 9836 controller and HP 4145 semiconductor parameter analyzer. The student undertaking this research also attended the SRC Technology Transfer Course at UC Berkeley concerning the BSIM model during October, 1985.

Plans

Activities scheduled for the next quarter include the development of a Programmers Manual for AIDE2, debugging the AIDE2 program as problems develop in its use to design and layout analog integrated circuits. A high level program is to be developed to interface with AIDE2 implementing a silicon compiler for analog-to-digital successive approximation converters or digital-to-analog converters. Consideration will be given to switching the design of the linear block from an n-well process to a p-well process and also to the development of blocks which can be implemented on the NSF/DARPA MOSIS facility. Other objectives include the completion of the partial simulation capability and the automated ability to physically replicate any simulation experimentally. The high level simulation program will be completed and the

problem of interfacing this program with lower level simulators will be investigated.

Publications and Presentations

Two presentations were given during this period at the 28th Midwest Symposium on Circuits and Systems held in Louisville, KY during August 1985. The first was a state of the art review on "CAD of Analog Integrated Circuits" and the second was titled "Development of a Synthesis Program for Successive Approximation Data Converters". The second publication was published on pages 305 to 307 of the Proceedings.

Personnel

The personnel involved with the project are Phillip Allen (Principal Investigator and Professor), Romeo Asibal (PhD student), Pat Barton (MS student), Desiree Ellsworth (MS student), Seong Hong (PhD student), John Parish (PhD student), Kwang Yoon (PhD student), and Bruce Young (MS student). Mrs. Sherri Brenner has been supported half-time as a staff assistant. Mrs. Brenner, Mrs. Ellsworth, Mr. Hong and Mr. Yoon have been supported by SRC during this quarter.

Expenditures

The expenditures for the period of July 1, 1985 through Nov. 30, 1985 are as follows:

	<u>July 1985</u>	<u>Aug. 1985</u>	<u>Sept. 1985</u>	<u>Oct. 1985</u>	<u>Nov. 1985</u>
Personal Services	4,859.25	4,342.68	13,882.13	4,854.52	4,329.48
Fringe Benefits	477.84	277.67	2,219.50	154.43	226.69
Materials & Supplies	347.22	324.15	1,121.27	207.70	584.11
Computer	-	-	-	-	-
Travel	-	-	445.51	-	-
Overhead	3,616.00	3,139.76	10,933.69	3,312.57	3,264.08
TOTALS	<u>9,300.31</u>	<u>8,084.26</u>	<u>28,602.10</u>	<u>8,529.22</u>	<u>8,404.36</u>

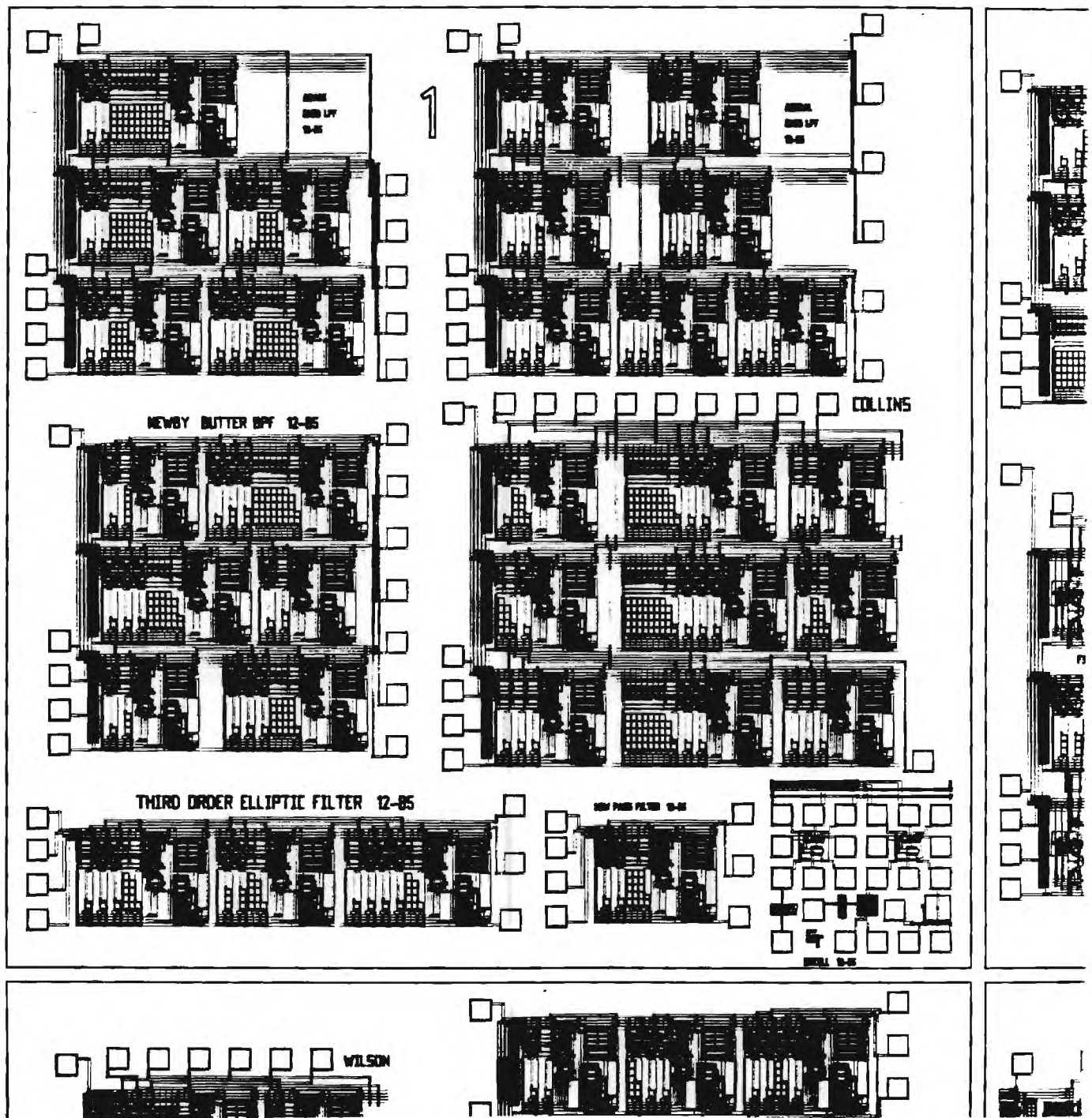


Figure 1 - Illustration of the upper left-hand quadrant of the multiple project chip containing AIDE2 designed circuits.

PRODUCED UNDER SRC CONTRACT? Yes/No/In Part SRC CONTRACT #: 84-07-051
OTHER SOURCES OF SUPPORT? Schlumberger Chair Funds and GT E-Funds

NAME OF PERSON RESPONSIBLE

Phillip E. Allen
Schlumberger Professor

NAME AND ADDRESS OF ORGANIZATION

School of Electrical Engineering
Georgia Institute of Technology
Atlanta, GA 30332

REPORT TITLE

Quarterly Report, December 1, 1985 to February 28, 1986

REPORT AUTHOR(S)

Phillip E. Allen

TYPE OF REPORT

TIME COVERED
(if applicable)

REPORT
DATE

PAGE
COUNT

DATE SUBMITTED
TO SRC

☒ Qtrly Progress
☐ Annual Report
☐ Final Report
☐ Thesis

FROM: 12/1/85
TO : 2/28/86

9/29/86

3

☐ Pub. Preprint - submitted to... date: _____
☐ Presentation - presented at.... date: _____

☐ Other - please describe.....

ABSTRACT OF CONTENTS OF THIS SUBMISSION (not to exceed 1 single spaced page)

This report discusses the progress made on the development of a multi-level analog integrated CAD program. Accomplishments are described in the areas of AIDE2 CAD program development and documentation, testing and testability of circuits designed by AIDE2, development of a high level analog simulator, and the development of precision small signal models for sub-micron devices. Major accomplishments include the design of circuits by AIDE2 for fabrication, documentation of the AIDE2 program, and the layout of a chip for fabrication of submicron devices for testing and modelling.

SUBJECT KEYWORDS (circle keywords supplied on reverse - note any additions)
LIST A: Analog, CAD, CMOS LIST B: IC/CAD Tools, auto. layout, dev. modeling

*** FOR SRC USE ONLY ***

DATE RECEIVED: _____ SRC PUBLICATION ID: _____



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

PHONE: (404) 894-6251

September 29, 1986

MEMORANDUM

TO: Ralph K. Cavin, III, Director Design Sciences, SRC

FROM: Phillip E. Allen, School of EE, Georgia Tech

SUBJECT: Quarterly Report, December 1, 1985 to February 28, 1986
SRC Project No. 84-07-051

Introduction

This report covers the period from December 1, 1985, to February 28, 1986, for the SRC sponsored research program at Georgia Institute of Technology titled, "A Multilevel Analog Integrated Circuit CAD Program." The objective of this research project is to continue the development of the AIDE2 CAD tool for analog integrated circuits, develop a multilevel-simulator for analog circuits, examine the development of testing methodology for the AIDE2 program, and to use the AIDE2 program at a high level. A second objective is to develop a precise, computer efficient, small signal model for submicron MOS devices suitable for analog integrated circuit design.

Accomplishments and Problems

This report will divide the research activities into three parts. These parts are research associated with the AIDE2 program, multilevel simulation of analog circuits, and precision, analog, small signal models for short channel MOS technology.

The AIDE2 program was applied last quarter to the design of integrated circuits, which are now in the process of being fabricated. During the present time, the User's Guide is about 50% complete and both p-well and n-well linear blocks have been incorporated in the cell library. A high level program called ADDAC, which uses AIDE2 to create the design and layout of an D/A converter of a successive approximation A/D converter, has revealed several bugs in the AIDE2 program. These bugs have been identified and removed. A masters thesis describing the ADDAC program has been completed [1]. A paper describing the ADDAC program will be presented at the 1986 Custom Integrated Circuits Conference [2].

Other research activities associated with the AIDE2 program include the development of AIDE2 on a PC. This effort has been hampered by the delay in obtaining the necessary hardware and software. The ability to simulate only a portion of the circuit designed by AIDE2 has been developed. The rest of the circuit is isolated from the simulated portion by using controlled sources which have a gain of unity or zero (for disconnection). The partial simulation capability is being developed as the input to the testing capability, which is being developed for AIDE2.

One of the important steps in the precision, analog, small-signal model research is the selection of BSIM as the large signal model to be used in finding the dc operating point. This model was found to give very good results when compared with experimental data for devices approaching 1 micron and in the weak inversion region. Accomplishments include the debugging and installation of the BSIM graphical program on our HP 9836, the development of routines to take the large signal information measured by the BSIM program and plot the small signal conductances, and the measurement of various MOS devices. The devices measurement include devices with L down to about 1.2 microns from General Electric and Silicon Systems. At this point the data is being accumulated with little interpretation until sufficient data exists.

Other accomplishments which support the above research are the adaptation and installation of the design rule checker CDRC which is suitable for a double-poly process, installation of SWITCAP on our VAX 11/780, and the development of the test facility to measure and characterize analog devices and circuits. It has been necessary to develop several programs for remote control of the data acquisition instruments.

Plans

The primary effort for the next quarter will be the preparation for the SRC Design Review and the AIDE2 Technology Transfer Course. This will require completion of the AIDE2 User's Guide, documentation of changes to and applications of AIDE2, completion of a high level simulation program for analog circuits and systems, the development of a physical test capability of AIDE2 circuits based on the partial simulation, evaluation of the short channel measurements and the development of routines for characterizing the data, and the redesign of the linear block suitable for the MOSIS CMOS technology.

Publications and Presentations

The only publication this quarter was by Pat Barton concerning his masters thesis describing the ADDAC program [1]. One paper was submitted for publication [2]. A presentation was made to General Electric, Schenectady, NY by Bruce Young on the AIDE2 program.

Personnel

The personnel involved with the project are Phillip E. Allen (Principal Investigator and Professor), Romeo Asibal (PhD student), Pat Barton (MS student), Desiree Ellsworth (MS student), Seong Hong (PhD student), John Parish (PhD student), Kwang Yoon (PhD student), and Bruce Young (MS student). Mrs. Sherri Brenner has been supported half-time as a staff assistant. Mrs. Brenner, Mrs. Ellsworth, Mr. Hong and Mr. Yoon have been supported by SRC during this quarter. Both Pat Barton and Bruce Young have completed their MS programs and have taken industrial positions. Pat will be employed at Schlumberger Well Services in Houston and Bruce will be employed at Silicon Systems in Tustin, CA.

Expenditures

	<u>Dec. 1985</u>	<u>Jan. 1986</u>	<u>Feb. 1986</u>	<u>Total</u>
Personal Services	\$4,025.79	\$1,222.43	\$12,670.23	\$17,918.45
Fringe Benefits	\$169.91	\$160.99	\$1,285.24	\$1,616.14
Materials & Supplies	\$431.96	\$345.90	\$1,285.24	\$898.16
Computer	-	-	-	-
Travel	\$340.90	-	-	\$340.90
Overhead	\$3,150.59	\$1,098.12	\$8,938.11	\$13,186.82
Totals	<u>\$8,119.15</u>	<u>\$2,827.44</u>	<u>\$23,013.88</u>	<u>\$33,960.47</u>

References

1. Pat R. Barton, "A Synthesis Program for CMOS Successive Approximation A/D and D/A Converters," Masters Thesis, School of Electrical Engineering, Georgia Institute of Technology, February, 1986.
2. P.E. Allen and P.R. Barton, "A Silicon Compiler for Successive Approximation A/D and D/A Converters," Proceedings of 1986 CICC, Rochester, NY, May 1986, pp. 552-555.

PRODUCED UNDER SRC CONTRACT? Yes/No In Part SRC CONTRACT #: 84-07-051
OTHER SOURCES OF SUPPORT? Schlumberger Chair Funds and GT E-Funds

NAME OF PERSON RESPONSIBLE

Phillip E. Allen
Schlumberger Professor

NAME AND ADDRESS OF ORGANIZATION

School of Electrical Engineering
Georgia Institute of Technology
Atlanta, GA 30332

REPORT TITLE

Quarterly Report, March 1, 1986 to May 31, 1986

REPORT AUTHOR(S)

Phillip E. Allen

TYPE OF REPORT

TIME COVERED
(if applicable)

REPORT
DATE

PAGE
COUNT

DATE SUBMITTED
TO SRC

- ☒ Qtrly Progress
☐ Annual Report
☐ Final Report
☐ Thesis

FROM: 03/1/86
TO : 05/31/86

10/06/86

12

- ☐ Pub. Preprint - submitted to... _____ date: _____
☐ Presentation - presented at.... _____ date: _____
☐ Other - please describe..... _____

ABSTRACT OF CONTENTS OF THIS SUBMISSION (not to exceed 1 single spaced page)

This report discusses the progress made on the development of a multi-level analog integrated CAD program. Accomplishments are described in the areas of AIDE2 CAD program development and documentation, testing and testability of circuits designed by AIDE2, development of a high level analog simulator, and the development of precision small signal models for sub-micron devices. Major accomplishments include the design of circuits by AIDE2 for fabrication, documentation of the AIDE2 program, and the layout of a chip for fabrication of submicron devices for testing and modelling.

SUBJECT KEYWORDS (circle keywords supplied on reverse - note any additions)
LIST A: Analog, CAD, CMOS LIST B: IC/CAD Tools, auto. layout, dev. modeling

*** FOR SRC USE ONLY ***

DATE RECEIVED: _____ SRC PUBLICATION ID: _____



GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

PHONE: (404) 894-6251

October 6, 1986

MEMORANDUM

TO: Ralph K. Cavin, III, Director of Design Sciences, SRC

FROM: Phillip E. Allen, School of EE, Georgia Tech

SUBJECT: Quarterly Report, March 1, 1986 to May 31, 1986
SRC Project No. 84-07-051: "A Multilevel Analog IC CAD Program"

Introduction

This report covers the period from March 1, 1986, to May 31, 1986, for the SRC sponsored research program at Georgia Institute of Technology titled, "A Multilevel Analog Integrated Circuit CAD Program." The objective of this research project is to continue the development of the AIDE2 CAD tool for analog integrated circuits, develop a multilevel simulator for analog circuits, examine the development of testing methodology for the AIDE2 program, and to use the AIDE2 program at a high level. A second objective is to develop a precise, computer efficient, small signal model for submicron MOS devices suitable for analog integrated circuit design. Bruce Young, who completed his MS this quarter, was employed by Silicon Systems.

Accomplishments and Problems

The primary activity during this quarter consisted of measuring the circuits fabricated at Harris Semiconductor and preparing for the Design Review and the Technology Transfer Course, which were held on June 16 and June 17, respectively. The major problem encountered was that the AIDE2 generated circuits did not work correctly because of two errors in the layout. A technique for directly measuring the small signal parameters of a MOSFET was developed in order to provide a means of evaluating the small signal model research.

During May, the integrated circuits designed by the AIDE2 program and fabricated by Harris Semiconductor were returned for testing and evaluation. Fig. 1 shows the entire chip layout. Fig. 2 shows a photograph of one of the fifth-order, low-pass filter which was designed as a benchmark filter (this circuit is the upper, right-hand corner of quadrant 1 of Fig. 1). The results of the testing uncovered two embarrassing layout errors. These errors can be seen by close examination of the photograph of Fig. 3 showing a first order filter (this circuit is at the bottom middle of quadrant 1 of Fig. 1). The first error was a reversal in the inputs to the op amp. This, of course, turned all negative feedback circuits into positive feedback. The second and more difficult error to locate was found in the transfer slice of the linear block. In the transfer slice, two inverters are used to create the inverse of ϕ_1 and ϕ_2 . It was found that one end of each inverter was not connected to the appropriate power supply causing the inverter to function on only half of the clock cycle.

The op amp connection error could and was repaired using microsurgery. However, the circuit still did not perform correctly, which led to the identification of the second error. It was not possible to correct the second error by microsurgery techniques. Consequently, the majority of the circuits on Fig. 1 unfortunately were useless. The op amp used in the linear block was tested and performed satisfactorily. The op amp had a gain of 4,615, an input voltage offset of 7.3mV, input common mode range of -4.3V to 4.7V, and an output swing (unity gain) of -5V to 4.7V. Test structures on the chip were also measured and more accurate electrical parameters were derived for the Harris process. These results are illustrated in Table 1.

Other activities during this quarter consisted of completing the AIDE2 Users Guide, connecting the KIC2 layout program with the SummaGraphics tablet resulting in speeding up the geometric description and editing of integrated circuits, measurement of capacitor ratios from the Harris chip, completion of an op amp design to be fabricated using the MOSIS facility, documentation of the HP equipment used in the IC test facility, and the development of an example illustrating the multilevel analog simulator.

Test Methodology

A test methodology has been developed which can be employed by the AIDE2 program to physically isolate all or parts of a system for experimental testing after fabrication. Once a system has been described to the AIDE2 program, simulation can be made of all or a part of the system. The ability to replicate experimentally any simulation has been developed and is being incorporated into AIDE2, taking advantage of the structured regularity of the program to provide a means of being able to physically isolate/access any port or combination of ports of a system. In the approach for simulation, a voltage-controlled, voltage source is used to simulate the switches which will be used to connect or disconnect the ports. The approach used in the layout uses a test cell which contains one stage of a shift register and a double throw switch. If simulation is to be replicated, the program inserts a test cell which will allow a physical reconfiguration to achieve the simulation by entering all the test cells.

Table 2 shows the accuracy using the VCVS switch model and the simulation running time for the 3rd and 5th order low-pass filters. Figs. 4 through 6 show the relationship of the degree of testability as a function of area. Fig. 4 shows a detailed layout of a third order filter with no testing capability. This layout takes approximately 5 minutes to generate the detailed plot. Fig. 5 shows the same filter with maximal testing capability. The area required is roughly 1.75 times that of Fig. 4. Fortunately, the designer would never want to maximally test a circuit. Fig. 6 shows the same filter with partial testing capability (only 2 blocks out of a total of 3 blocks). The area required in this case is about 1.16 times that of Fig. 4. The time required to generate these detailed layouts was 6 minutes for Fig. 5 and 5.2 minutes for Fig. 6. These times are based on the AIDE2 implemented on a VAX 11/780 with the UNIX 4.2 operating system.

Precision Small Signal Analog Models for Submicron MOSFETs

Five new routines have been implemented into the BSIM extraction program, which include IDS vs. VBS, GDS vs. VDS, GM vs. VGS, $\log(GM)$ vs. VGS, and GMB vs. VBS. These subroutines are shown in Fig. 7. Two techniques were used to generate the small signal conductances in each small signal routine. The first one is the numerical differentiation technique which produce the small signal conductance data based upon the DC measurement data from HP 4145A. The second one is the analytical differentiation approach which derives the small signal conductance model equations from BSIM drain current equations. Since the true small signal conductances (which should be measured by the ac measurement technique) were not available, the test structure shown in Fig. 8 has been developed to measure the true small signal conductances at a given bias point. Fig. 9 shows the comparison of three small signal conductance measuring techniques for a NMOS device with $W/L = 20\mu m/20\mu m$. These three techniques will be applied to devices with channel lengths down to $1.2\mu m$ in order to develop an accurate small signal conductance model. Unfortunately, smaller length devices are not easily available.

Plans

The primary activity for the next quarter will consist of the design review, technology transfer seminar, and the correction of the errors in the AIDE2 layout. Measurements from the short channel devices will be used to develop possible small signal models, which give the desired precision. Because of summer activities and plans, only three students (Seong Hong, Danny Shamlou, and Kwang Yoon) will be working during the summer quarter.

Publications and Presentations

Three presentations concerning the experience and knowledge developed on the subject of analog circuit design methodology and CAD techniques were given during this quarter. The first was a presentation titled, "Analog VLSI Design Tutorial," given at the presymposium workshop on Analog and Digital VLSI given in conjunction with the 1986 IEEE International Symposium on Circuits and Systems, held in San Jose, CA on May 4, 1986. Two papers were presented at the 1986 IEEE Custom Integrated Circuits Conf. at Rochester, New York on May 14, 1986 [1,2].

Personnel

The personnel involved with the project are Phillip E. Allen (Principal Investigator and Professor), Romeo Asibal (PhD student), Pat Barton (MS student), Danny Shamlou (MS student), Seong Hong (PhD student), John Parish (PhD student) and Kwang Yoon (PhD student). Mrs. Sherri Brenner has been supported half-time as a staff assistant. Mrs. Brenner, Mr. Shamlou, Mr. Hong, and Mr. Yoon have been supported by SRC during this quarter. Other individuals (Romeo Asibal and Juvena Loo) have been supported from other sources, but have been working on SRC research.

Expenditures

Personal Services	\$6,808.00	\$6,607.00	\$5,321.00	\$18,736.00
Fringe Benefits	\$180.00	\$169.00	\$256.00	\$605.00
Materials and Supplies	\$291.00	\$100.00	\$256.00	\$916.00
Travel	-	-	\$377.00	\$377.00
Computer	-	-	-	-
Overhead	\$4,622.00	\$4,366.00	\$4,114.00	\$13,102.00
<hr/>	<hr/>	<hr/>	<hr/>	<hr/>
TOTALS	\$11,901.00	\$11,242.00	\$10,593.00	\$33,736.00

References

1. P.E. Allen and P.R. Barton, "A Silicon Compiler for Successive Approximation A/D and D/A Converters," Proc. of IEEE 1986 Custom Integrated Circuits Conference, Rochester, NY, May 12-15, 1986, pages 552-555.
2. P.E. Allen, "A Tutorial - Computer Aided Design of Analog Integrated Circuits," Proc. of IEEE 1986 Custom Integrated Circuits Conference, Rochester, NY, May 12-15, 1986, pages 608-616.

Table 1 - GTCELL Results.

GTCELL MEASUREMENTS
(R. L. Asibal---May 20, 1986)

SSI-IVA DEVICE PARAMETERS

PARAMETER	HARRIS'	MEASURED VALUE		
		LOW	MEDIUM	HIGH
NMOS:				
Vto	0.7 Volt	0.70 Volt	0.71 Volt	0.72 Volt
Conduction (UoN*Cox)	35uA/V**2	20.03uA/V^2	28.3uA/V^2	28.85uA/V^2
Lambda	(not given)	.0069/V	.0077/V	.0115/V
		LOW	MEDIUM	HIGH
PMOS:				
Vto	-0.7 Volt	-0.65 V	-0.65 V	-0.65 Volt
Conduction (UoN*Cox)	12uA/V**2	9.87uA/V^2	10 uA/V^2	10.33uA/V^2
Lambda	(?)		0.0121/V	
Sheet Resistivities				
		LOW	MEDIUM	HIGH
Poly I	20 Ohm/sq	15.06	15.46	
Poly II	100 Ohms/sq		73.37	
P +MOAT			24.44	
P + Diffusion	40 Ohms/sq			
P Well			27.056	

Measurements of the above parameters were done using HP4145 Semiconductor Parameter Analyzer and using the programs PLOT and VTOPLLOT. The Program VTOPLLOT obtains measurement data of drain currents of NMOS or PMOS transistor and takes its SQUARE ROOT value. This is useful when trying to extract the VTO values.

SIMULATION

• Order		3rd order	5th order
• VSVC model	VDB (%)	0	0
	accuracy VP (%)	0	0
$\text{accuracy} = \{ (" \text{manual} " - " \text{program} ") / " \text{manual} " \} * 100 (\%)$			
• Run time without test (sec)		142.59	669.67
• Run time with test (sec)		157.77	688.73

Table 2. Accuracy and simulation running time

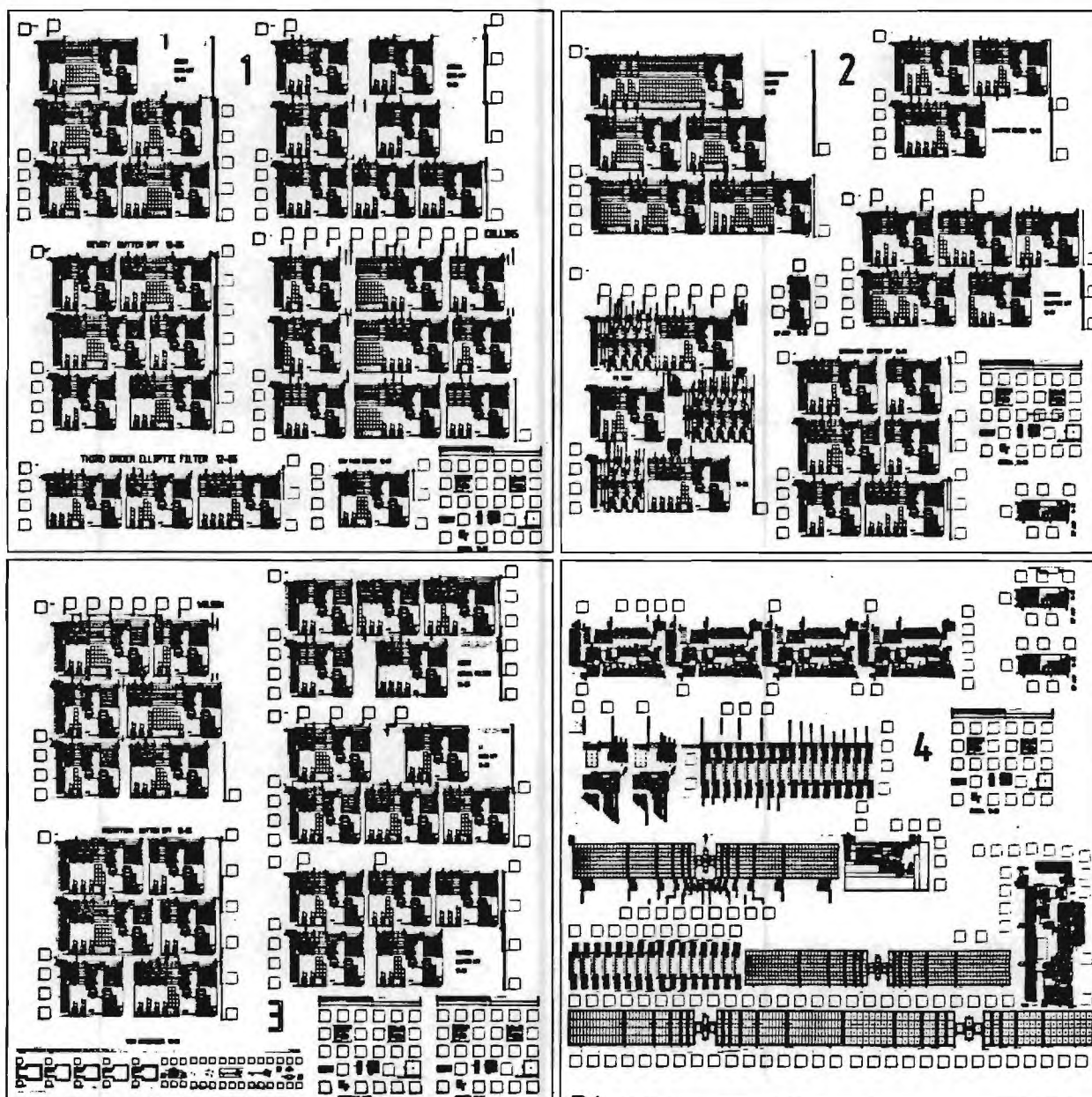


Fig. 1 - Computer layout of the Harris Multiple Project Chip No. 1.

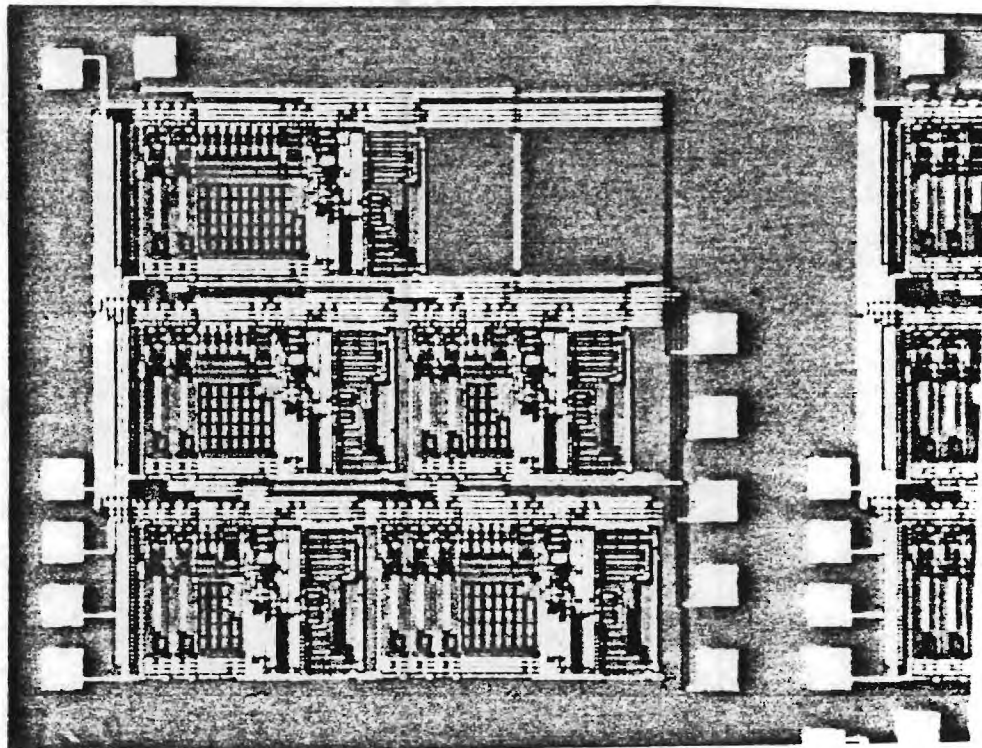


Fig. 2 - Photograph of the fifth-order, low-pass benchmark filter of Fig. 1.

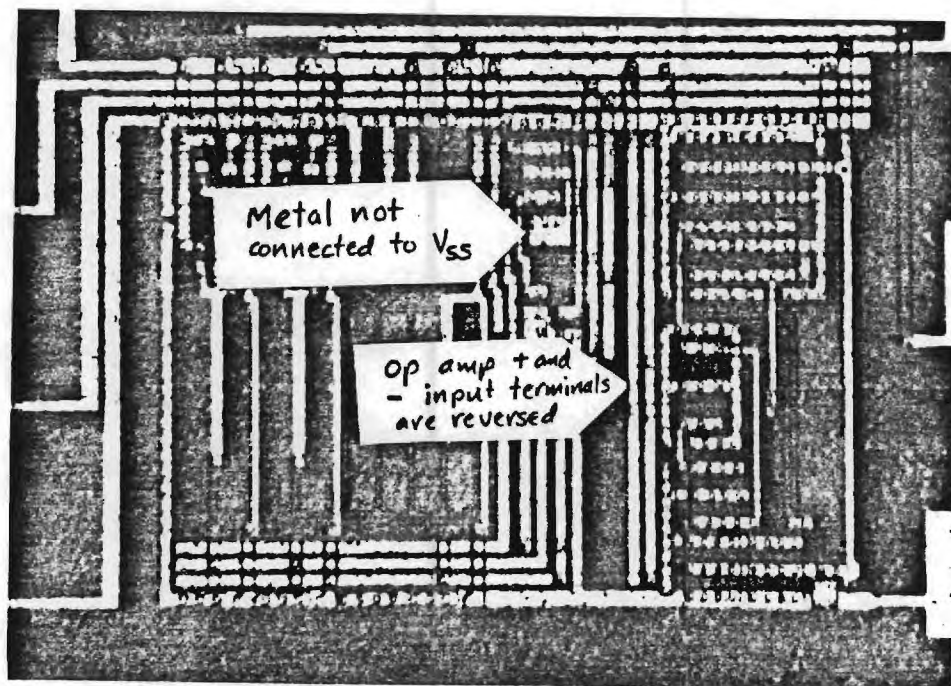
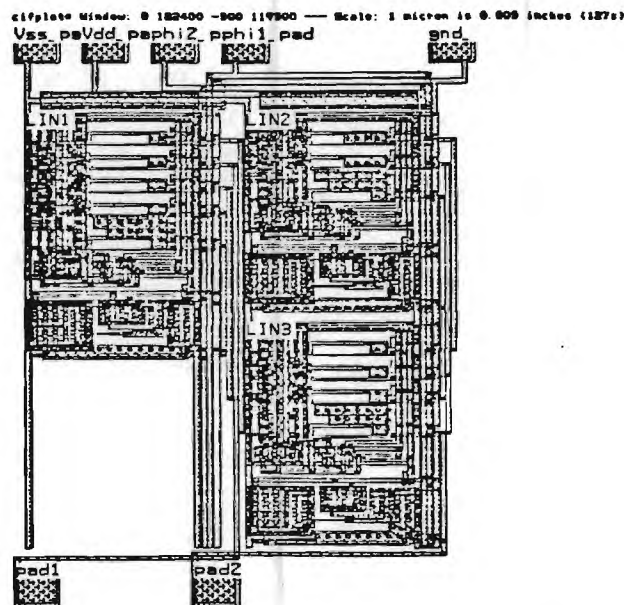
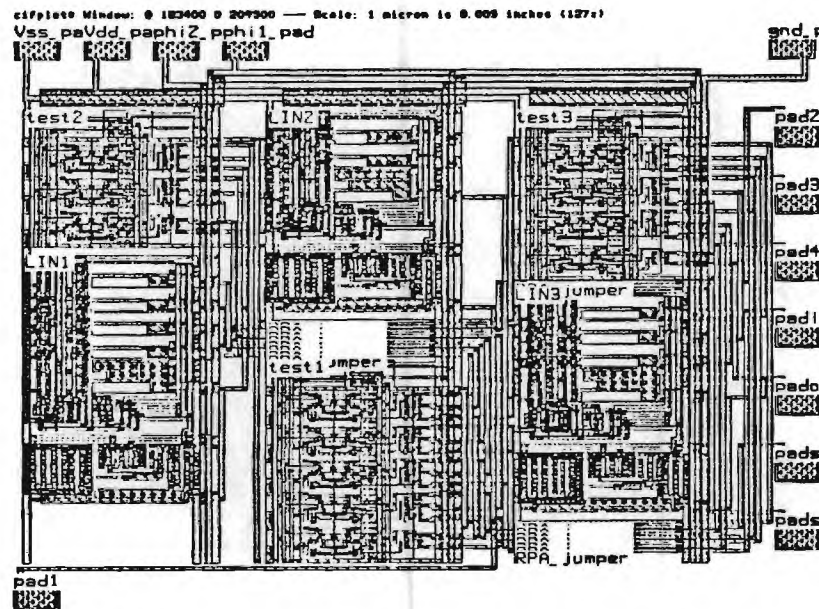


Fig. 3 - Photograph of the first-order linear block illustrating the errors found in debugging.



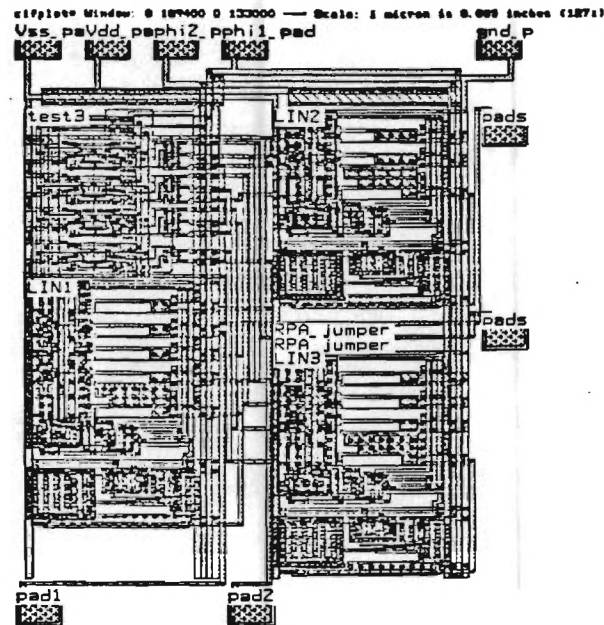
$$\text{AREA} : 1824 \mu \times 1195 \mu = 2179580 \mu^2$$

Fig. 4 - Third-order filter layout with no testing capability.



$$\text{AREA} : 1834 \mu \times 2093 \mu = 3838562 \mu^2$$

Fig. 5 - Third-order filter layout with maximal testing capability.



$$\text{AREA} : 1894 \mu * 1330 \mu = 2519020 \mu^2$$

Fig. 6 - Third-order filter layout with partial testing capability.

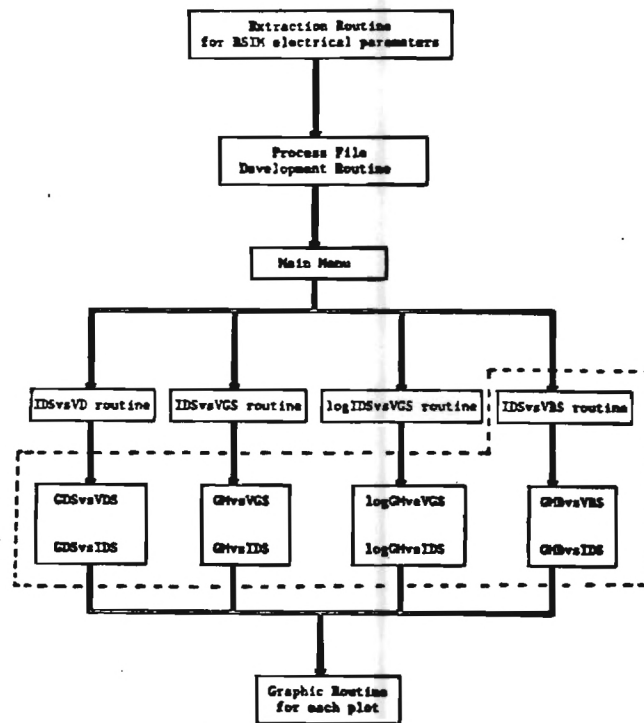
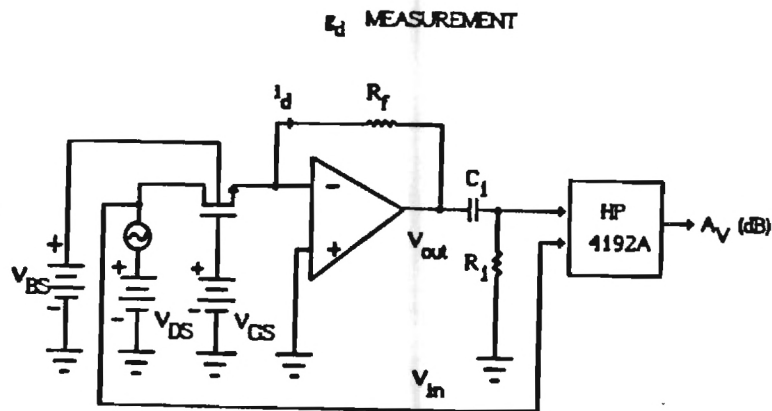


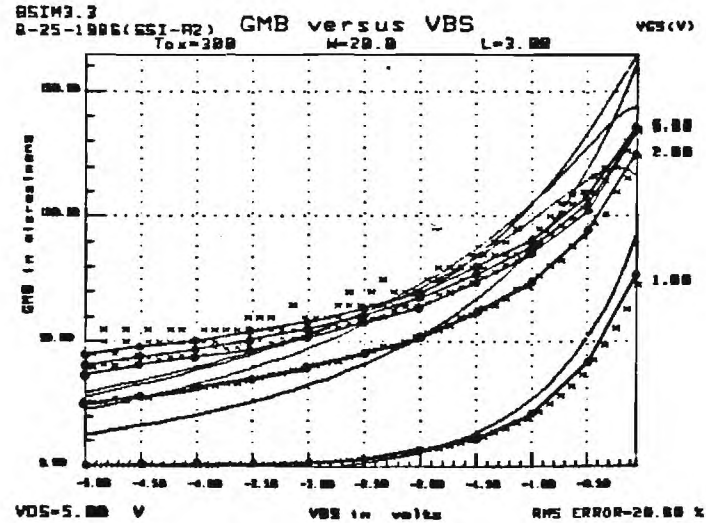
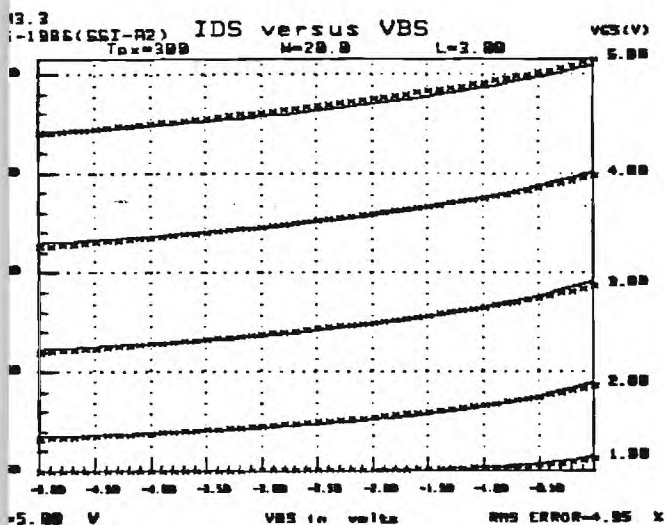
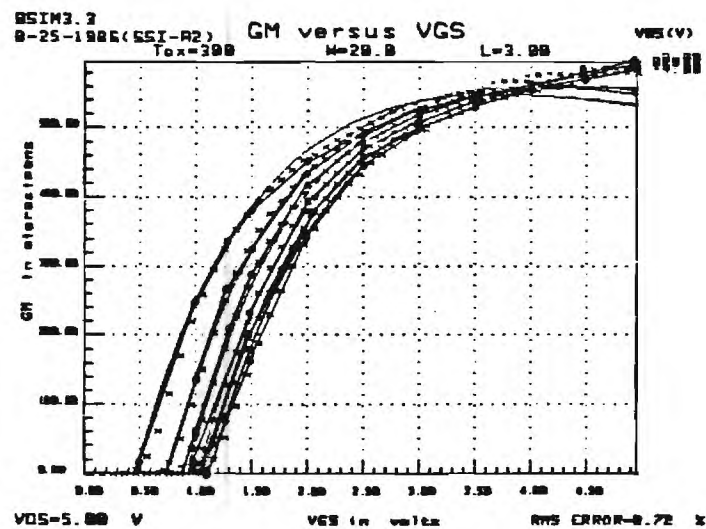
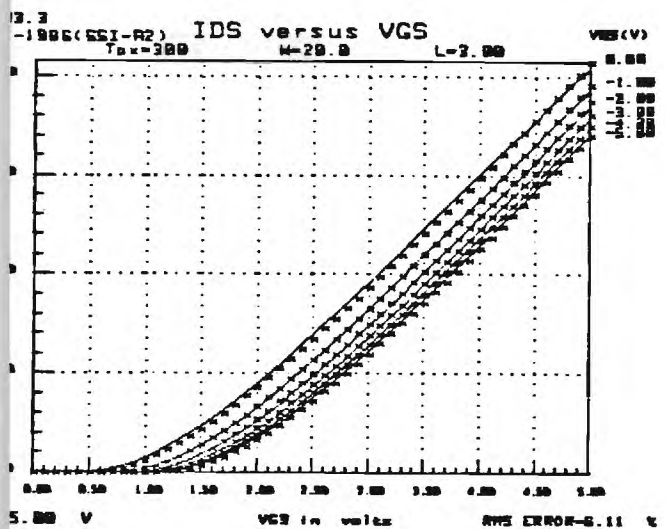
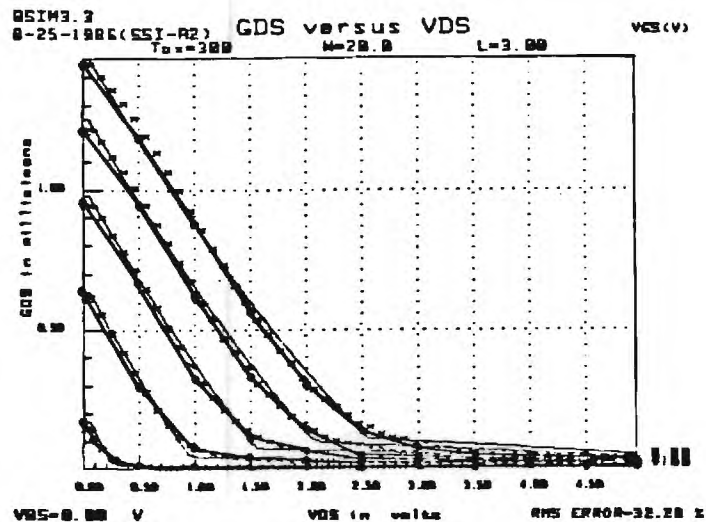
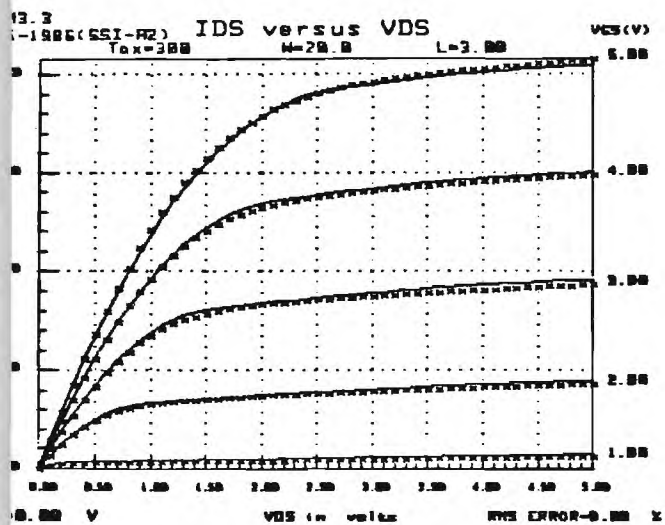
Fig. 7 - Modified BSIM program for the development of the small signal models



$$A_V = 20 \text{ LOG} (V_{out} / V_{in}) = 20 \text{ LOG} \left(\frac{i_d R_f}{V_{in}} \right) = 20 \text{ LOG} (g_d R_f)$$

$$\therefore g_d = \frac{i_d}{V_{in}} = \frac{1}{R_f} 10^{(A_V / 20)}$$

Fig. 8 - AC direct measurement technique for small signal conductances



X:Numerical approach — :Analytical approach —●— : AC measurement

Fig. 9 - The comparison of three techniques for W/L=20μm/3μm N-MOSFET device

SRC REPORT/PUBLICATION SUBMISSION FORM

Please complete this form for all reports and publications submitted to SRC.

PRODUCED UNDER SRC CONTRACT? Yes/No/ <u>In Part</u>		SRC CONTRACT #: <u>84-07-051</u>		
OTHER SOURCES OF SUPPORT? <u>Schlumberger Chair</u> Funds and GTE-Funds				
NAME OF PERSON RESPONSIBLE		NAME AND ADDRESS OF ORGANIZATION		
Phillip E. Allen Schlumberger Professor		Georgia Institute of Technology School of Electrical Engineering Atlanta, Ga. 30332-0250		
REPORT TITLE				
Annual Project Report - A Multilevel Analog Integrated Circuit CAD Program				
REPORT AUTHOR(S)				
Phillip E. Allen				
TYPE OF REPORT	TIME COVERED (if applicable)	REPORT DATE	PAGE COUNT	DATE SUBMITTED TO SRC
<input type="checkbox"/> Qtrly Progress				
<input checked="" type="checkbox"/> Annual Report	FROM: <u>Sept. 1984</u>	<u>April 3, 1986</u>	<u>?</u>	<u>?</u>
<input type="checkbox"/> Final Report	TO : <u>August 1985</u>			
<input type="checkbox"/> Thesis				
<input type="checkbox"/> Pub.Preprint - submitted to ...			date: _____	
<input type="checkbox"/> Presentation - presented at ...			date: _____	
<input type="checkbox"/> Other - please describe				
ABSTRACT OF CONTENTS OF THIS SUBMISSION (not to exceed one single spaced page)				
<p>This report describes the progress made in the development of CAD tools for analog integrated circuit design. A program designated AIDE2 has been developed to define, simulate and layout analog integrated circuits at a high, moderate, and/or low level. The moderate interface with the AIDE2 program is through a C-language circuit description program. The high level interface requires the user to provide a high level synthesis program which will take the specifications of the circuit or system and convert them to a C-language circuit description program. The low level use of AIDE2 is the manner in which new blocks are added to the library and requires circuit design and programming expertise.</p> <p>The AIDE2 program has been used to design and submit integrated circuits for fabrication. Two silicon compilers designed around AIDE2 are described. These compilers design switched capacitor filters and A/D or D/A successive approximation converters, respectively. The physical structure of the design methodology employed in AIDE2 has permitted the development of methods of physically testing the circuits designed by AIDE2. This research has shown that it is possible to use CAD techniques to design analog integrated circuits. The next issue of concern is how do these circuits compare with custom designed analog integrated circuits.</p>				
SUBJECT KEYWORDS (circle keywords supplied on reverse - note any additions)				
* * * F O R S R C U S E O N L Y * * *				
DATE RECEIVED: _____		SRC PUBLICATION ID: _____		

ANNUAL PROJECT REPORT

SRC Contract No. 84-07-051

April 3, 1986

A MULTILEVEL ANALOG INTEGRATED CIRCUIT CAD PROGRAM

Prepared for

Dr. Ralph K. Cavin, III
Manager, Design Sciences Program
Semiconductor Research Corporation
300 Park Drive
P.O. Box 12053
Research Triangle Park, NC 22709

by

Phillip E. Allen
School of Electrical Engineering
Georgia Institute of Technology
Atlanta, GA 30332

A MULTILEVEL ANALOG INTEGRATED CIRCUIT CAD PROGRAM

Abstract

This report describes the progress made in the development of CAD tools for analog integrated circuit design. A program designated as AIDE2 has been developed to define, simulate and layout analog integrated circuits at a high, moderate, and/or low level. The moderate interface with the AIDE2 program is through a C-language circuit description program. The high level interface requires the user to provide a high level synthesis program which will take the specifications of the circuit or system and convert them to a C-language circuit description program. The low level use of AIDE2 is the manner in which new blocks are added to the library and requires circuit design and programming expertise.

The AIDE2 program has been used to design and submit integrated circuits for fabrication. The results of the fabrication are not yet available for evaluation. Two silicon compilers designed around AIDE2 are described. These compilers design switched capacitor filters and A/D or D/A successive approximation converters, respectively. The physical structure of the design methodology employed in AIDE2 has permitted the development of methods of physically testing the circuits designed by AIDE2. Documentation of the AIDE2 program includes a Users Manual and several conference publications as well as SRC publications. This research has shown that it is possible to use CAD techniques to design analog integrated circuits. The next issue of concern is how do these circuits compare with custom designed analog integrated circuits.

TABLE OF CONTENTS

1.0	Introduction.....	1
2.0	Organization of AIDE2.....	3
3.0	Design Interfaces with AIDE2.....	4
3.1	Intermediate Interface.....	5
3.2	High Level Interface.....	11
3.3	Low Level Interface.....	12
4.0	Simulation of AIDE2 Circuits.....	16
5.0	Layout and placement of AIDE2 Circuits.....	18
6.0	Use of AIDE2 for Intermediate Level Design.....	25
7.0	Use of AIDE2 for High Level Design.....	30
7.1	AROMA - An Analog Silicon Compiler for SC Filters.....	30
7.2	An Analog Silicon Compiler for A-D Converters.....	34
8.0	Fabrication of AIDE2 Circuits.....	40
9.0	Testability of AIDE2 Circuits.....	42
10.0	Technology Aspects of AIDE2.....	44
11.0	Summary and Plans.....	46
12.0	References.....	48

1.0 INTRODUCTION

This report describes the progress made in the development of a multilevel computer-aided design (CAD) program which can be used to simulate and layout analog integrated circuits using a CMOS technology. The objective is to increase analog IC design productivity, insure the ability to test the resulting design, and to be compatible with CAD techniques for digital IC design [1]. This research is a result of earlier considerations on how such a CAD methodology might be applied to the design of analog integrated circuits [2].

Automation has been provided for the component layout, circuit synthesis, and system layout phases of the IC design process. Three basic concepts utilized include module compilers to generate the primitive components, soft macros which are specified in the C programming language, and automatic placement and routing for system layout. These features permit the programmability necessary for analog circuits where the capacitor values, transistor sizes, and the clock phasing are not known until after the design process.

The result of this research has been the development of a CAD program designated as AIDE2. This program provides the CAD tools to design integrated circuits at the circuits and systems level [3]. Fig. 1 illustrates the inputs and outputs of the program. The output of the program is a simulation input file and a complete layout. The program has been used to design switched capacitor circuits at both the circuit level [4] and system level [5]. A benchmark circuit is in the process of fabrication along with other circuits whose purpose is to evaluate the performance of AIDE2 as a design tool for analog integrated circuit design. The AIDE2 program has been distributed to

interested SRC members and universities. Two MS theses have resulted from the work described in this report [6,7]. Present activities include debugging, documentation, and the inclusion of more cells within the internal library of AIDE2.

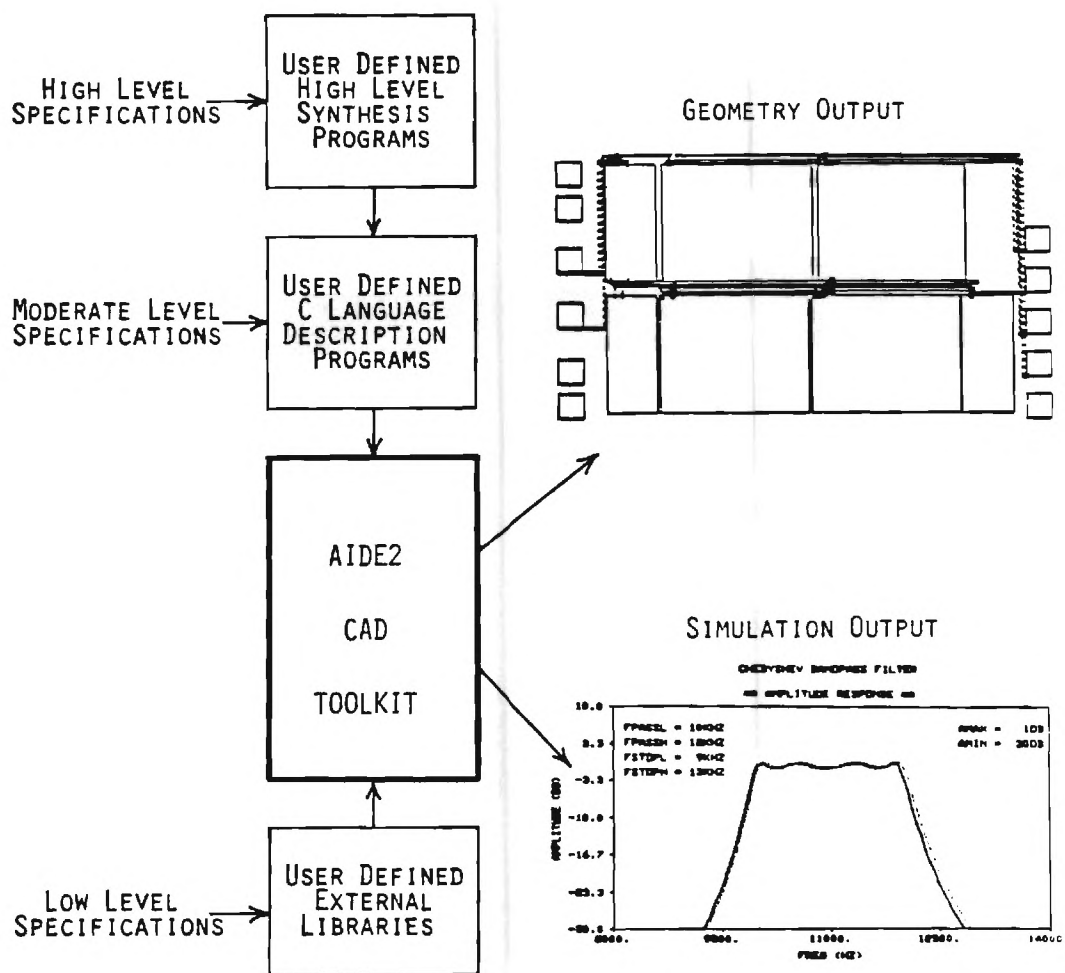


Fig. 1 - Illustration of the input levels and outputs of AIDE2.

This report will focus on the AIDE2 program and its development. The organizational viewpoint of the program will be considered first. Next, the various levels at which the user can interface with the program will be described. The primary outputs of the AIDE2 program, namely a simulation file and a layout, will be discussed next. Several examples of using the AIDE2

program at the circuits level where the interface is a C-language description of the circuit will be presented. Examples of using the program at a higher level will be illustrated. The high level examples are essentially analog silicon compilers for a limited class of analog systems. Finally, the progress of efforts to establish a test methodology for the AIDE2 program will be described. Future effort and work under progress will also be presented in this report.

2.0 ORGANIZATION OF AIDE2

Fig. 2 shows how the AIDE2 program is organized. In the AIDE2 program, C language software functions are available for defining the two main parts of the description and the chip layout. The circuit description part is specified using unique subroutine calls to particular parameterized and standard cells that are stored in the cell library facility of AIDE2. The hierarchical organization of the circuit description is such that the lowest level calls point to fixed or parameterized cells and the arguments of these calls simply define the netlist information and the cell's characteristics. This

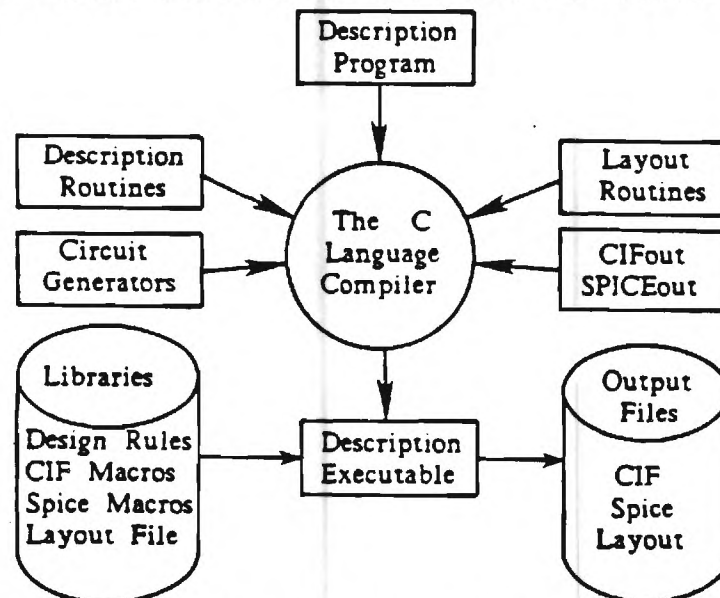


Fig. 2 - AIDE2 organization.

arrangement of calling into service certain combinations of parameterized and standard cells is quite useful in the synthesis of various configurations of system level analog circuits. AIDE2 uses a procedural circuit design language in which circuits are described by procedural calls in a C language program. This program, referred to as a description program, consists of 2 parts, the circuit description and the layout control which are executed sequentially.

The circuit description section specifies the netlist of the circuit. Each C language subroutine represents a cell or a macro of cells. The layout control section calls subroutines which perform a standard cell layout. At the end of the layout, the CIF_out routine is called to output the circuit geometries in the CalTech intermediate format (CIF) [8]. As the circuit description executes, the input simulation files for SPICE [9] are generated.

The description program may be written as a single circuit description or as an interactive design system which generates a circuit description resulting in circuit synthesis. The circuit layout procedure is in the form of a standard cell layout in which I/O ports are automatically connected to pads for bonding or brought out to the edge of the layout area.

3.0 DESIGN INTERFACES WITH AIDE2

There are three distinct levels at which the design can interface with the AIDE2 program. These levels are illustrated in Fig. 1 and are the high level, intermediate level, and the low level. These levels are also called the systems, circuits, and devices level, respectively. The intermediate level interface with AIDE2 is through a C language interface referred to as a circuit description program. The high level interface is accomplished through a user-defined synthesis program which takes certain circuit/system specifications and

automatically generates a circuit description program. The low level interface is essentially the method by which new cells are entered into the AIDE2 library. These cells may be fixed or parameterized and must conform to the layout constraints of AIDE2.

3.1 Circuit Description at the Intermediate Level

At the intermediate level of design, it is assumed that the cell library facility of AIDE2 contains the necessary cells for design. These cells may be fixed, parameterized, or macros. All cells are represented in the circuit description by subroutine calls.

Fixed geometry cells are represented by the same subroutine called `StandardX()` in which the name to the cell and the nets connected to the cell are passed as arguments to the subroutine. The value of `X` is equal to the number of port connections. For example, `Standard1("pad","pad5",n7)` is the call used to create a pad named "pad5" to be connected to port 7 of the circuit.

Parameterized cells are represented by their own unique subroutine. The name of the subroutine is the same as the name of the cell. Arguments to the routine specify nets connected to the cell and any parameters needed to customize the cell (i.e. capacitor values, switch phasing, etc.). An example of a parameterized block used for switched capacitor circuits will be discussed shortly.

Macros are also represented by their own unique subroutine and have the same types of arguments as parameterized cells and look the same as parameterized cells in the circuit description program. Physically, however, macros represent an interconnection of cells which may be fixed, parameterized or other macros. Macros may be hierarchically nested to any level.

An example will be considered to illustrate the principles of the intermediate design level. Fig. 3 shows the general configuration of a linear block used to design switched capacitor filters. For this block, the user must specify the clock frequency which is common to all the building blocks, the input and transfer switch phasing for every input node of each block, the node identification number, and the capacitor value associated with each input node. Table 1 shows the circuit element description of the linear block of Fig. 3.

The assignment protocol for this block will now be considered. Referring to the example of Fig. 3, the user may arbitrarily assign a number 1 to the topmost node, n_2 , and its associated circuit elements, then assign the number 2 to node, n_3 , and all the circuit elements associated with it, and so on. The output node is numbered last, in this case, 7. The input switch connected with node n_2 and node n_5 are arbitrarily assigned the input switch phasing of 2 while those connected with nodes n_3 and n_4 are arbitrarily assigned an input switch phasing value of 1. If there is no switch connected between a node and a capacitor as in node n_1 and capacitor C_5 , the input switch phasing value is

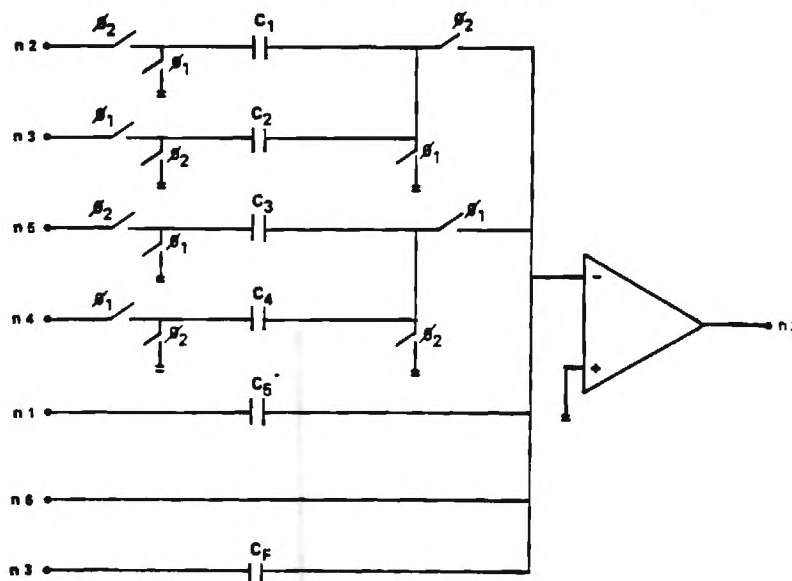


Fig. 3 - General configuration of the linear block.

3. The transfer switch phasing connected with C_1 and C_2 is assigned a value of 2 while that connected with C_3 and C_4 is assigned a value of 1. There is no transfer switch between capacitor C_5 or C_f and the inverting input of the op amp so for this case the transfer switch value is 3. Input node, n6, is directly connected to the op amp inverting input so the input switch phasing value is 4 and the transfer switch phasing value is not needed. The out_net for this example is n3 and ninputs is 7 because there are a total of 7 input nodes (including the output node). The user may define up to 10 nets per linear block. Table 1 summarizes the circuit element description of the linear block of Fig. 3.

Input Number	Node	Input Switch Phasing	Transfer Switch Phasing	Capacitor Value
1	n2	2	2	2.235
2	n3	1	2	2.754
3	n5	2	1	3.215
4	n4	1	1	1.893
5	n1	3	3	2.084
6	n6	4	-	0.0
7	n3	3	3	2.756

Table 1 - Circuit element description of the linear block of Fig. 3.

For the linear block parameterization indicated in Table 1, the description format is: out_net = n3 ; ninputs = 7

```

in_nets[1] = n2 ; in_nets[2] = n3 ; in_nets[3] = n5 ;
in_nets[4] = n4 ; in_nets[5] = n1 ; in_nets[6] = n6 ;
in_nets[7] = n3 ;
in_phase[1] = 2 ; in_phase[2] = 1 ; in_phase[3] = 2 ;
in_phase[4] = 1 ; in_phase[5] = 3 ; in_phase[6] = 4 ;
in_phase[7] = 3 ;

```



```

t_phase[1] = 2 ; t_phase[2] = 2 ; t_phase[3] = 1 ;
t_phase[4] = 1 ; t_phase[5] = 3 ; t_phase[6] = 3 ;
cap[1] = 2.235 ; cap[2] = 2.754 ; cap[3] = 3.215 ;
cap[4] = 1.893 ; cap[5] = 2.084 ; cap[7] = 2.756

```

In the example that follows, three linear blocks are used to build the third-order, low-pass, elliptic filter shown in Fig. 4. The C language description

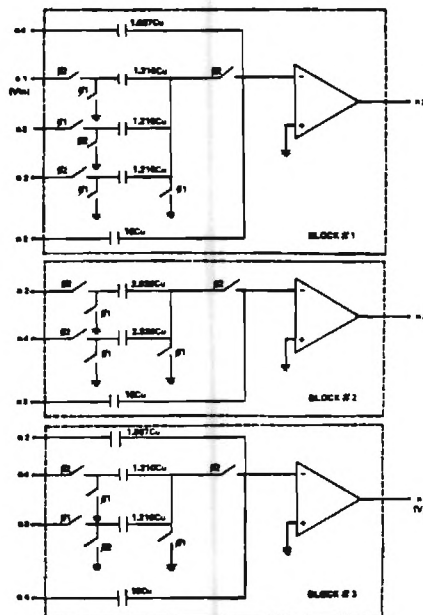


Fig. 4 - Third-order, low-pass, elliptic, switched capacitor filter.

program for the third-order filter is shown in Fig. 5. The details of this are

```

#include <stdio.h>
#include <math.h>
#include "usr1/allen/yong/aide2/lib/all.ext"
main() /* 3rd order elliptic low-pass filter */
{
/*
Filter circuit description:
*/
Begin4("lp3fil",n1,n2,n3,n4)
{
/* Declaration of the variables that describe the linear block */
int out_net,ninputs,in_nets[10],in_phase[10],t_phase[10];
int plotype=0;

```

(Fig. 5 - Continued on next page)

```

float fc,cap[10]
char spice_file[20], cif-file[20];
/* Get name of spice file: */
printf("\nEnter name of SPICE data file: ")
scanf("%s",spice_file);
/* Initialization of spice routine */
fc = 100e3; /* Defines clock frequency of 100KHz */
begin_ckt(spice-file,"3RD ORDER ELLIPTIC LP FILTER",fc);

/*
Block #1
*/
out_net = n2; ninputs = 5;
in_nets[1] = n4; in_nets[2] = n1; in_nets[3] = n3;
in_nets[4] = n2; in_nets[5] = n2; /* Defines input nodes */
in_phase[1] = 3; in_phase[2] = 2; in_phase[3] = 1;
in_phase[4] = 2; in_phase[5] = 3; /* Defines input switch
phasing for capacitors */
t_phase[1] = 3; t_phase[2] = 2; t_phase[3] = 2; t_phase[4] = 2;
t_phase[5] = 3; /* Defines trans. sw. phasing of capacitors */
cap[1] = 1.087; cap[2] = 1.210; cap[3] = 1.210;
cap[4] = 1.210; cap[5] = 10.00;
linear_block("LIN1",spice_file,out_net,ninputs,in_nets, in_phase,
t_phase,cap,fc); /* This statement will create a SPICE deck for
this block. It will also load the necessary parameters to create
an instance of the block in the circuit description. The para-
meters may be used to create a complete layout of the circuit. */

/*
Block #2
*/
out_net = n3; ninputs = 3;
in_nets[1] = n2; in_nets[2] = n4; in_nets[3] = n3;
in_phase[1] = 2; in_phase[2] = 2; in_phase[3] = 3;
t_phase[1] = 2; t_phase[2] = 2; t_phase[3] = 3;
cap[1] = 2.926; cap[2] = 2.926; cap[3] = 10.00;
linear_block("LIN2",spice_file,out_net,ninputs,in_nets,in_phase,
t_phase,cap,fc);

/*
Block #3
*/
out_net = n4; ninputs = 4;
in_nets[1] = n2; in_nets[2] = n4;
in_nets[3] = n3; in_nets[4] = n4;
in_phase[1] = 3; in_phase[2] = 2;
in_phase[3] = 1; in_phase[4] = 3;
t_phase[1] = 3; t_phase[2] = 2; t_phase[3] = 2; t_phase[4] = 3;
cap[1] = 1.087; cap[2] = 1.210; cap[3] = 1.210; cap[4] = 10.00;
linear_block("LIN3",spice_file,out_net,ninputs,in_nets,in_phase,
t_phase,cap,fc);
/* The following interconnects the blocks in the SPICE deck. */
printf ("\n interconnect routine :");

```

(Fig. 5 - Continued on next page)

```

interconnect(spice-file);
printf ("\n input_source routine :"); /* This routine writes the
input source statement in the SPICE file. */
input_source(spice_file,"IN",n1,2,0.0,1.0)
printf("\n anastat routine :"); /* This routine obtains the type of
SPICE analysis and the parameters from the user. */
anastat(spice_file);
printf ("\n output routine :");
output(spice_file,"AC",n4,1);
printf ("\n end_ckt routine :");
end_ckt(spice_file);
/* The following statements create pads that are connected to the
circuit. */
Standard1("pad","pad1",n1); /* Input pad to the circuit. */
Standard1("pad","pad2",n4); /* Output pad of the circuit. */
/* The following statement perform the analog standard cell
layout of the circuit. */
Analog_layout(50)
/* The following statements get the CIF filename and determine the
type of plot. */
printf("\n Enter name of CIF data file: ");
scanf("%s",cif_file);
printf("\n Enter a '1' for a bounding box plot or");
printf("\n      a '0' for a detailed plot :);
scanf("%d",&plotype);
/* The following statement generates a CIF file from the layout
information. */
CIF_out(cif_file,plotype);
}
End();
}

```

Fig. 5 - C language description program for the 3rd-order filter of Fig. 4.

found in the Users Manual [10], however, some of the subroutines which are pertinent at this point are described in the following.

The BEGIN(Circuit Name) statement always occurs at the start of the circuit description and allows the hierarchy of the circuit description to be maintained throughout the chip layout. The statement "int ni;ni = define_net(net_name)" defines a net to be used in the circuit description and also assigns a net value to the net declared. The net is associated with a net name which is a string of up to 14 characters. If there are less than 10 local nets used, the Begin X macro may be used instead of the Begin and Define_net

procedure. The `BeginX()` macro will execute the `Begin()` statement, define the nets with the netnames the same as the variable name, and declare the local net variables as integers. This subroutine call is used in lieu of the `Begin` and `Define_net` calls described above if less than 10 nets are required for the linear block. The value of `X` ranges from 2 to 10.

The `Standard_cell("Cell_name", instancename, n1, n2, ..., n10)` procedure calls a standard cell and fits it into the current circuit design. It also assigns the cell an arbitrary instance name and updates the net list according to the nets at its I/O port locations. The `StandardX()` macro simplifies the standard cell procedure call so that the user does not have to write ten I/O port values when the cell has less than ten ports. Finally, the `End()` statement ends the circuit description program. Usually, each circuit description program is a C language subroutine with the subroutine name being the same as the circuitname in the `Begin()` statement.

3.2 Use of AIDE2 for High Level Design

The system designer or novice user who does not wish to write a C language circuit description may design circuits with AIDE2 through specific interactive design systems. In order to accomplish this objective, it is necessary to develop a high level synthesis program capable of taking a high level circuit or systems description and translating this description into a C language circuit description which can be input to the AIDE2 program. Once the require cells are placed within the AIDE2 cell library, such a program is not difficult to write.

The high level program depends upon the input. In some cases, the "high level" program can be an interactive program which saves the designer from having to type every line of Fig. 5. On the other extreme, the high level

program will take system specifications and convert these specifications into a circuit description program. Such programs may have built-in intelligence and may make decisions based upon the information inputted by the designer. These high level programs when interfaced with AIDE2 form an analog silicon compiler for limited classes of analog circuits and systems. Two such programs have been written and will be discussed in Sec. 7.0 when the high level use of AIDE2 to design circuits and systems will be considered [5,11].

3.3 Use of AIDE2 for Low Level Design

The low level use of AIDE2 is considered to be the process of entering new cells into the AIDE2 cell library. Because the designer has total control of the circuit design and its layout, this level would correspond to case when the designer is not satisfied with existing blocks and therefore must define his own. In order to enter new cells, the designer must be familiar with the database organization of AIDE2. The designer must also be familiar the C programming language and the CalTech Intermediate Format (CIF).

AIDE2 is a technology independent design system in which the layout design rules are stored in a file called the token file. The token file contains all of the C language tokens (C macros) which are expanded into the AIDE2 software by the C language preprocessor at compile time. For this reason, AIDE2 is technology independent after compilation. A different cell library should be created for each incompatible technology. The technology used for the present library cells is a typical 3 to 5 micron, double-poly, CMOS technology. More information on this technology is found in Appendix A.

The organization of AIDE2 and the files it reads are illustrated in Fig. 6. The circuit is specified in the circuit description program. The description program is compiled and linked to the AIDE2 library and layout

routine which are stored in the object files io.o and lay.o, respectively. The AIDE2 source code, called io.c and lay.c, references the token file. The reference is a #include statement which specifies the directory pathname of the token file. More information concerning the organization of AIDE2 can be found in the references [6].

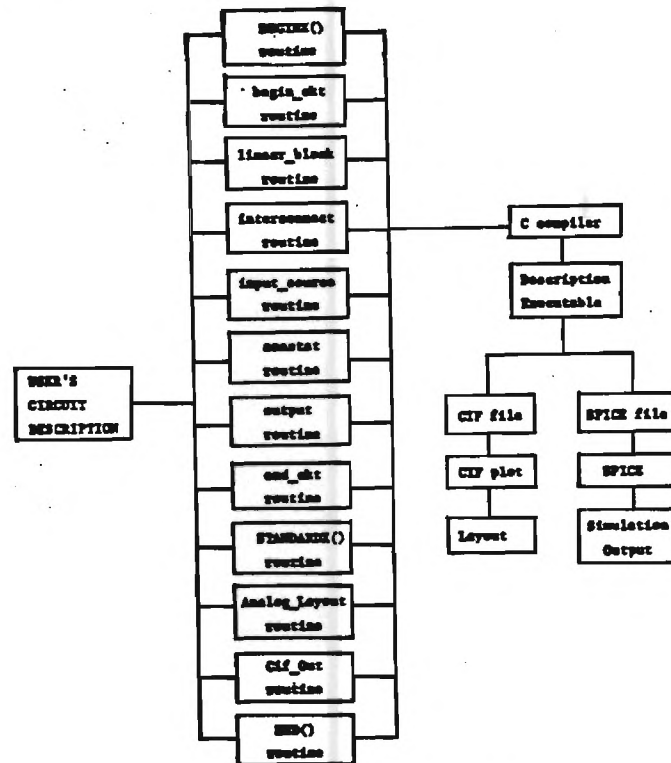


Fig. 6 - Organization of the AIDE2 subroutines.

Fixed geometry standard cells are stored in individual files. The Standard_cell() routine, which is in the io.c library, is used to specify an instance of a cell in the netlist. Optionally, the instance may be specified with the StandardX() routine which provides a more terse format. The StandardX() routines are C macros which are expanded into Standard_cell() calls by the C preprocessor. The StandardX() macros are defined in the token file. This means that instances of all fixed geometry standard cells are actually specified by the Standard_cell() routine.

The `Standard_cell` routine finds the standard cell definitions by opening a file called `standard.cif`. `Standard.cif` contains a simple cross-reference of each standard cell name with the directory and filename that the cell is stored in. Multiple standard cells may be stored in the same file if they have unique symbol numbers. Cells in different files may have any value for symbol numbers. All cells may be defined hierarchically to any level.

The design of standard cells is straight forward. There are basic CIF records and various CIF extensions which are unique to AIDE2. More details, information, and examples on this procedure can be found in the Users Manual [10].

Parameterized cells are the second type of cells which used by AIDE2. Parameterized cells provide much greater flexibility than fixed geometry cells. However, they are more difficult to design. The details of the design procedure is beyond the level of this report and can be found elsewhere [12]. The two types of cells compare as follows:

1. The location of a fixed geometry cell is specified in the `CELL_LAY.DAT` file. The location of parameterized cell geometries is specified in the customizing software for that cell.
2. An instance of a fixed geometry cell is specified by the `StandardX()` routine. An instance of a parameterized cell is specified by calling the customizing software for that cell. Each parameterized cell has its own customizing software. Library parameterized cells are currently stored in the `pcell.c` file which is compiled and lined with the circuit description program. However, these cells may be stored in any file.

Aside from these differences, the design of both types of cells is similar and the final layout of a parameterized standard cell should meet the same

AIDE2 standard cell design rules which hold for fixed geometry cells. The AIDE2 parameter preprocessor could be used to design any shape cell if the AIDE2 standard cell layout is not to be used.

The organization of the parameter preprocessor is illustrated in Fig. 7. The customizing software may be stored in the pcell.c file which would be one of the library I/O routines in the figure. When the customizing routine is called, it invokes various other AIDE2 I/O routines. These routines store the parameter data (in memory) and copy the cell abstractions to the layout data. The layout routines will use the layout data for placement and routing. The CIF_out() routine will use the parameter data to expand the cell for the final CIF file.

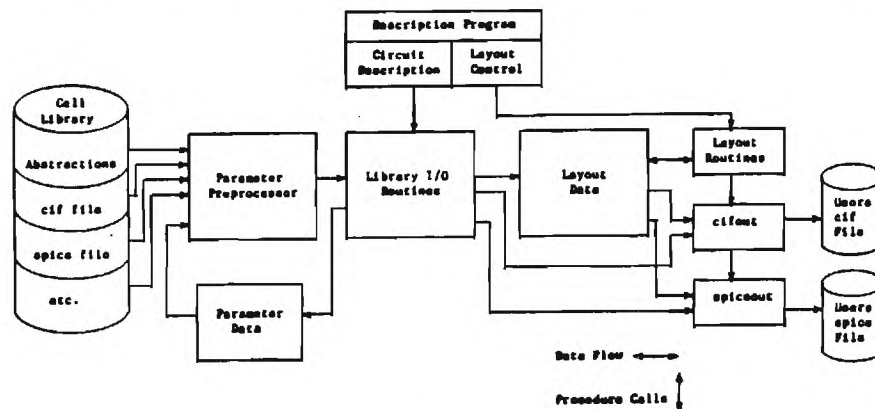


Fig. 7 - Organization of the AIDE2 parameter preprocessor.

This section has covered the design interfaces that are presently available to the AIDE2 program. The primary interface is the C language description program of the circuit or system. A high level program is used to automatically or interactively generate this program for a given circuit or system. The low level interface not only requires the ability to design the analog circuit cell but an understanding of the organization of the AIDE2 program and its database structure. More details can be found in the AIDE2 Users Manual [10] and the AIDE2 Programmers Manual [12].

4.0 SIMULATION OF AIDE2 CIRCUITS

One of the important functions of any CAD design tool is to provide the user with the electrical verification of the design. For this reason, the circuit description can contain procedural calls which will yield an input file for simulation. At the present, only SPICE2 input files are generated. The designer must write the necessary procedural statements that will generate the various control cards normally present in any SPICE file. The resulting input file will be in a form ready to apply to a SPICE simulator.

The analyses available include all the analysis capabilities of SPICE. It is the responsibility of the designer of each library cell to provide a simulation block routine which is appropriate for the desired simulation. The simulation block routine will create a SPICE subcircuit which is placed in the input simulation file. The simulation block routine is a C language function call to invoke parameterized cells from the cell library and is called after the `begin_ckt` routine which is described in the following. The function of the simulation block routine is to generate a complete SPICE subcircuit description for each instance of the parameterized block. It is called by the statement:

```
sim_block("object_name",spice_file,out_net,ninputs,in_nets,parameter1,
          parameter2,...,parameterX)
```

An example of this routine applied to the linear block of Fig. 3 is

```
linear_block("LIN1",spice_file,out_net,ninputs,in_nets,in_phase,
             t_phase,cap,fc)
```

where the arguments of the routine have been defined in Section 3.1. An example of a generated file as a product of an execution of the linear block would be,

```

.SUBCKT "obj_nam"
+991 992      ! output nodes (991 => phi1, 992 => phi2)
+ 11 12      ! input node 1 (11 => phi1, 12 => phi2)
+ 21 22      ! input node 2 (21 => phi1, 22 => phi2)
.
.
.
+ N1 N2      ! input node N (N1 => phi1, N2 => phi2)
.
.
.
(SPICE description of the block)
.
.
.
.ENDS "obj_name"

```

If the block is parameterizable, this simulation block routine can be complex. Further details of writing the block simulation routine can be found in the Programmers Manual [12]. The most often used simulation for switched capacitor networks is the frequency domain. This can be accomplished with SPICE as indicated in the above example by creating a circuit for phase 1 and another for phase 2 and using the transmission line feature to simulate a delay unit [13].

Some of the routines used to complete the simulation file are described as follows. The `begin_ckt` routine is used to open the user's SPICE file and write the user supplied title at the top of the file. If simulating a discrete-time circuit, the clock frequency f_c must be included with the `spice_file` and the title in the argument of `begin_ckt`. The `interconnect` routine is activated after all simulation block routines (i.e. parameterized cells such as the linear block of Fig. 3) have been called. This routine generates the subcircuit call statements and performs the interconnections between the various circuit building blocks. The `input_source` routine connects a voltage source to the user's circuit for simulation purposes. This routine implements the connection of a voltage source with a dc value and an ac value to the input

net of the circuit at a specified phase of the clock. If the simulation block is a continuous-time circuit, then the clock phase should not be specified. The anastat routine is used to prompt the user for the type of analysis desired. The choices for the analysis of the linear block of Fig. 3 are ac or dc signal analysis. However, the analysis chosen must agree with that specified in the output() subroutine which immediately follows in the C description program. The output routine will create the output statement for SPICE. The output statement is the .PLOT or .PRINT statement and their arguments. The end_ckt routine causes a .END statement to be placed at the end of the simulation description.

5.0 LAYOUT AND PLACEMENT OF AIDE2 CIRCUITS

The two primary outputs of AIDE2 are the simulation described in Section 4.0 and the layout and placement which is described in this section. AIDE2 has adopted a standard cell layout system which provides high reliability and very fast turnaround. The floorplan of the cells are shown in Fig. 8. The standard

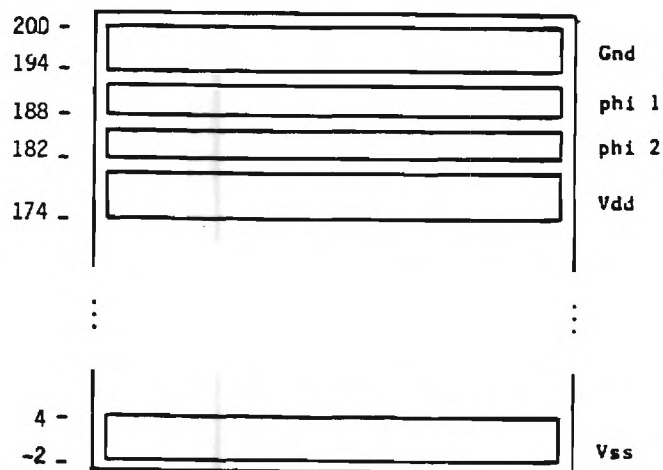


Fig. 8 - Floorplan for analog cells of AIDE2 circuits.

cells stored in the cell library facility have fixed heights of 202 lambda units and variable widths. The input/output (I/O) ports are placed vertically on polysilicon 1 and the busses are placed horizontally on metal. These busses are located within each cell and include V_{SS} , V_{DD} , Gnd, Phil, and Phi2. These busses have been selected primarily for switched capacitor applications. It is important to keep the analog signals distinct from digital signals in order to avoid undesired effects such as coupling and noise. The clocks were designed to be bussed rather than routed in order to provide more control on their interaction with analog signals. The ground buss is located at the top of the cell. Below the ground buss is the Phil and Phi2 busses followed by the V_{DD} buss. The V_{SS} buss is located at the bottom of the cell. The space between the V_{DD} and V_{SS} busses is available for the circuit. Unfortunately, the vertical efficiency of the cell in terms of circuit area to the total area is 84%. It is possible for the designer to place non-metal circuits under the busses which would increase this efficiency.

The chip floorplan has been designed in such a way that cells are placed in rows and are separated by routing channels. Fig. 8 shows the floorplan of the AIDE2 program. The external part of the floorplan have been reserved for pads. The lower four pads of the left pad area (3) are designated as the V_{SS} , V_{DD} , Phil, and Phi2 going from the bottom up. The ground pad is the highest pad of the left pad area (3). V_{SS} is connected directly by metal to the V_{SS} busses while V_{DD} crosses under the V_{SS} metal in order to contact the V_{DD} busses. The cross under is done in diffusion and has a width nearly equal to the height of the cell resulting in very little extra resistance. The cross unders for the clocks are not important since the clocks all see only gates of MOS devices which is a high impedance load. The number of pad placement areas,

i.e. the right pad area (4), top pad area (5), and the bottom pad area (6), depend upon the number of external connections required by the circuit.

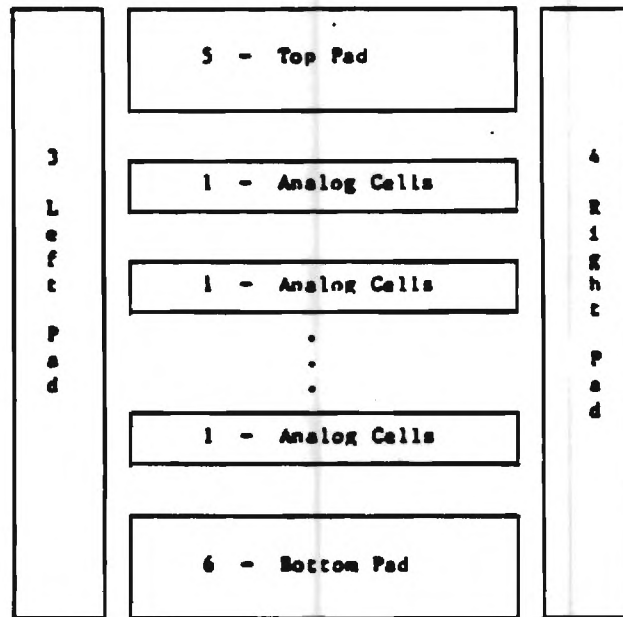


Fig. 9 - AIDE2 chip floorplan.

The standard cell system adopted by AIDE2 is equipped with placement and routing software. Therefore, placement and routing are done automatically. The placement program assigns the various cells to specific placement areas and controls the relative ordering of cells. The routing program on the other hand defines the paths of interconnections between the I/O ports of each cell in the circuit. The salient electrical characteristics and geometric information of each cell are stored in the cell library. When the user specifies an instance of a cell, the program execution of the AIDE2 description file will automatically read the abstraction information which contains all the geometric data needed for a layout. The abstractions also contain the symbolic information such as the cell name and directory of filenames for simulation subcircuits. When the user invokes the CIF_out routine, the CIF primitives

representing the cell's abstractions are copied directly into the user's output CIF file. The C language function call to the CIF_out routine is

```
CIF_out(CIF_filename,plot_type).
```

This routine will generate a CIF file from the layout information and prompts the user for the name of the CIF_file. The argument plot_type is 0 for a detailed plot and 1 for a bounding box plot. The bounding box plot is useful when the user wants to see the layout without the detail of every level of the CIF file.

The circuit layout in AIDE2 is provided with a control function that allows the user to select the optimum shape of the chip floorplan. Normally a square dimension is the best shape and this can be obtained by specifying an aspect ratio of 100. Higher values of this ratio will produce more rows that are shorter while a lower value will result in fewer but wider rows of cells. The calling statement for the circuit layout aspect ratio is

```
Analog_layout(aspect ratio).
```

Fig. 10 shows the bounding box layout of a circuit containing 16 large cells and 12 small cells for an aspect ratio of 100. Fig. 11 shows the layout of the same number and type of cells for an aspect ratio of 50. Fig. 12 shows the same number and type of cells for an aspect ratio of 200. The algorithm used to generate the circuit layout given the aspect ratio is not continuous in nature. It must take in account the number of rows and routing requirements resulting in discrete jumps in the actual aspect ratio. The ordering of the cells in the description program can also influence the layout of an AIDE2 designed integrated circuit. Between varying the order of the cells and the aspect ratio, the user can generally shape the floorplan as desired. It should be noted that because of extra routing required, the non-square layouts of the

same circuit typically require more area.

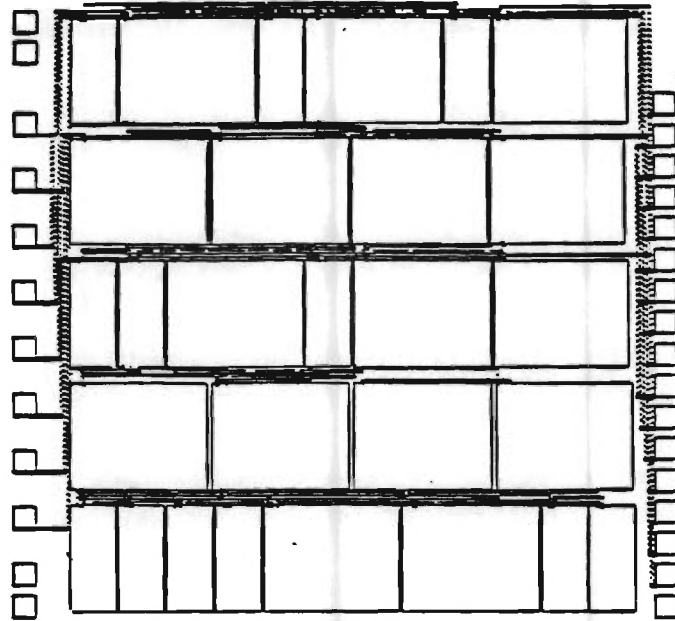


Fig. 10 - Test circuit layout with an aspect ratio of 100.

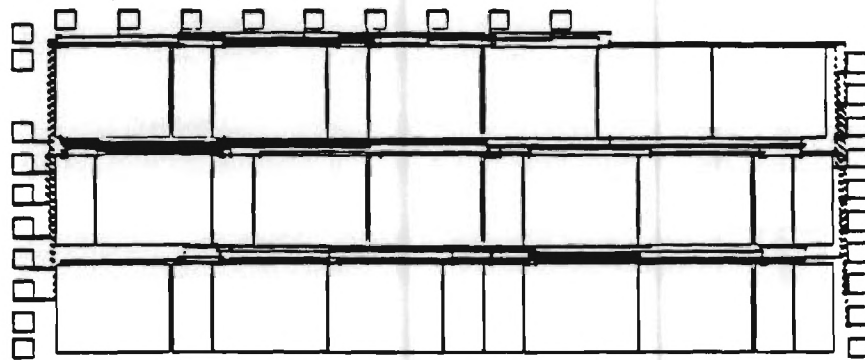


Fig. 11 - Test circuit with an aspect ratio of 50.

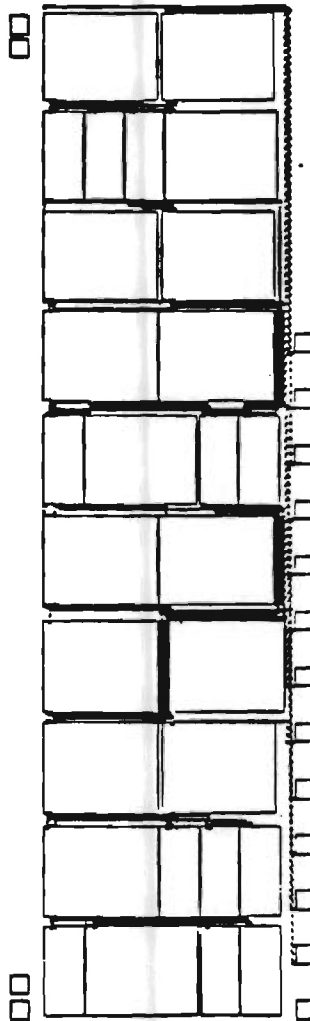


Fig. 12 - Test circuit with an aspect ratio of 200.

The design of the software layout program in AIDE2 was done in a top-down process and initially partitioned into the following tasks.

1. Floorplanning of the placement areas, routing channels and busses.
2. Relative placement of objects.

3. Global routing of nets.
4. Coordinate assignment of objects, placement and channel areas.
5. Estimate of channel sizes.
6. Channel routing of each channel.

The floorplanning determines the number of rows required which is dependent on two variables: the total area the objects will occupy and the aspect ratio for the chip. The number of rows (num) and the width of the standard placement areas (RPA width) are calculated from the following.

$$\text{total_width} = \sum_{i=1}^{\text{OBJnum}} \text{std_OBJ_width} \quad (1)$$

$$\text{num} = (\text{aspect_ratio}/100) \sqrt{\text{total_width}/\text{std_height}} \quad (2)$$

$$\text{RPA_width} = \text{total_width}/\text{num} \quad (3)$$

The placement program assigns objects to placement areas and specifies the relative order of the objects with the area. It does not specify the coordinates of the objects because this information is not needed. Either digital or analog objects may be placed. The standard cell areas must alternate from channels to placement areas and point to each other from top to bottom. Any of the rows may be either digital or analog placement areas and the routine is called once for each type. As in floorplanning, chip placement is done in two steps: standard cell and pad placement.

Routing is the process of defining paths which interconnect the ports in each net of a circuit. Most routing algorithms break the problem down to a sequence of point-to-point connections by dividing each net with n pins into $(n-1)$ two-point connects. To route a chip, channels must be defined and the interconnections of pins in the nets must be broken down into a sequence of routing the connections with each channel. The second step is called global

routing. The global routing algorithm of AIDE2 consists of two parts. The first is the scan of horizontal channels and the second is the pad global routing in the vertical channels. The channel routing algorithm used in AIDE2 is called a left edge algorithm [14]. More details on the layout algorithms used in AIDE2 can be found in one of the references [6].

6.0 USE OF AIDE2 FOR INTERMEDIATE LEVEL DESIGN

The background for using the AIDE2 program at the intermediate level has been covered in the last three sections. In this section, an example of using AIDE2 to design a switched capacitor filter will be presented. Fig. 13 shows

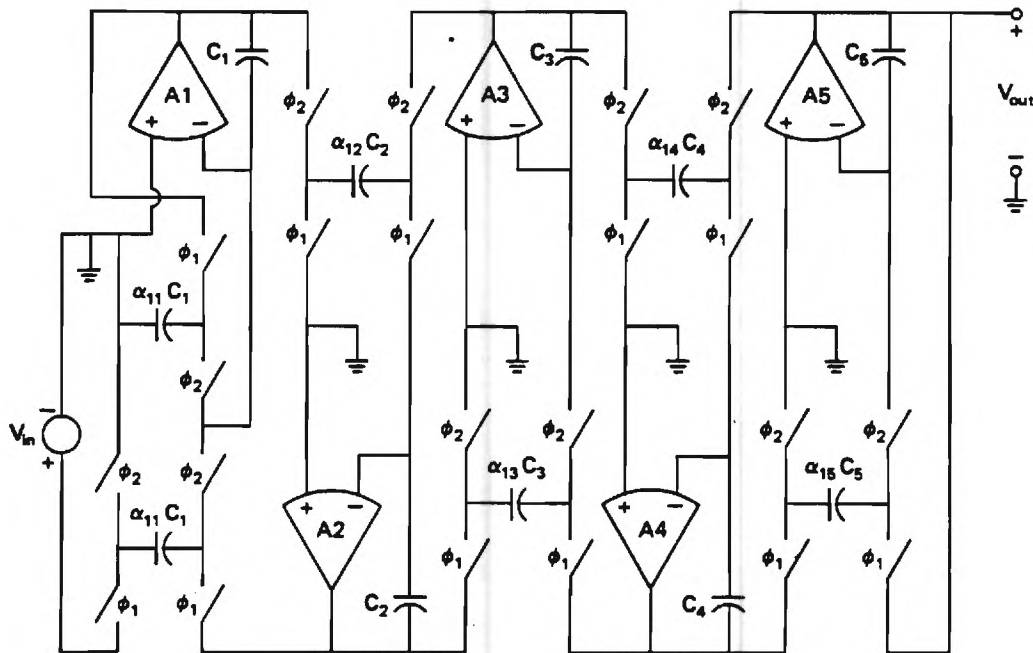


Fig. 13 - A 5th-order, switched capacitor, Chebyshev, low-pass filter.

the schematic diagram of a 5th-order, Chebyshev, 1dB ripple, low-pass filter. The circuit uses five linear blocks connected by six nets designated as n1, n2, n3, n4, n5, and n6. The input is connected to n1 during phase 1 while the output is to be observed at n6 during phase 2. The capacitor values are given

in terms of ratios which are $\alpha_{11} = \alpha_{21} = 0.02943$, $\alpha_{12} = 0.05759$, $\alpha_{13} = 0.02094$, $\alpha_{14} = \alpha_{24} = 0.05759$, and $\alpha_{15} = 0.02943$. The circuit is designed for a cutoff frequency of 1 KHz for a clock frequency of 100 KHz.

The description program written in C language is given in Fig. 14. The circuit description starts with the Begin6() macro statement since 6 nets are used. The required parameter variables are then declared and initialized. The linear block parameters are initialized prior to each call to the linear block routine. Afterwards, the remaining simulation and layout procedural statements are called.

```
#include <stdio.h>
#include <math.h>
#include "usr/usrsrc/aide2/lib/all.ext"
main() /* 5th order Chebyshev Low-pass Filter */
{
    /*
        Filter Description
    */
    Begin6("lp5fil",n1,n2,n3,n4,n5,n6)
    {
        /* Declaration of the variables that describe the linear block */
        int out_net,ninputs,in_nets[10],in_phase[10],t_phase[10];
        int plotype=0;
        float fc,cap[10];
        char spice_file[20], cif_file[20];
        /*
            Get name of Spice File:
        */

        printf("\nEnter name of SPICE data file: ")
        scanf("%s",spice_file);
        /* Initialization of SPICE Routine */
        fc = 100E3; /* Defines the clock frequency of 100 KHz */
        begin_ckt(spice_file,"5th Order Chebyshev LP Filter",fc);

    /*
        Block #1:
    */
        out_net = n2; ninputs = 4; in_nets[1] = n1; in_nets[2] = n3
        in_nets[3] = n2; in_nets[4] = n2; /* Defines input nodes */
        in_phase[1] = 1; in_phase[2] = 2; in_phase[3] = 2;
        in_phase[4] = 3; /* Defines the input switch phasing for each input */
        t_phase[1] = 2; t_phase[2] = 2; t_phase[3] = 2; t_phase[4] = 3;
        /* Defines the transfer switch phasing for each input */
        cap[1] = 1; cap[2] = 1; cap[3] = 1; cap[4] = 33.98
        linear_block("LIN1",spice_file,out_net,ninputs,in_nets,in_phase,t_phase,cap,fc);
        /* This call will create a SPICE input file for this block and load the
            necessary description parameters which are used to create a layout */

    /*
        Block #2
    */
        out_net = n3; ninputs = 3; in_nets[1] = n2; in_nets[2] = n4; in_nets[3] = n3;
        in_phase[1] = 2; in_phase[2] = 1; in_phase[3] = 3;
        t_phase[1] = 1; t_phase[2] = 1; t_phase[3] = 3;
        cap[1] = 1; cap[2] = 1; cap[3] = 17.45;
        linear_block("LIN2",spice_file,out_net,ninputs,in_nets,in_phase,t_phase,cap,fc);

    /*
        Block #3
    */
        out_net = n4; ninputs = 3; in_nets[1] = n3; in_nets[2] = n5; in_nets[3] = n4;
        in_phase[1] = 1; in_phase[2] = 2; in_phase[3] = 3;
        t_phase[1] = 2; t_phase[2] = 2; t_phase[3] = 3;
        cap[1] = 1; cap[2] = 1; cap[3] = 47.76;
        linear_block("LIN3",spice_file,out_net,ninputs,in_nets,in_phase,t_phase,cap,fc);
    }
}
```

(Fig. 14 - Continued on the next page)

```

      Block #4
*/
out_net = n5; ninputs = 3; in_nets[1] = n4; in_nets[2] = n6; in_nets[3] = n5;
in_phase[1] = 2; in_phase[2] = 1; in_phase[3] = 3;
t_phase[1] = 1; t_phase[2] = 1; t_phase[3] = 3;
cap[1] = 1; cap[2] = 1; cap[3] = 17.36;
linear_block("LIN4",spice_file,out_net,ninputs,in_nets,in_phase,t_phase,cap,fc);
/*
      Block #5
*/
out_net = n6; ninputs = 3; in_nets[1] = n5; in_nets[2] = n6; in_nets[3] = n6;
in_phase[1] = 1; in_phase[2] = 2; in_phase[3] = 3;
t_phase[1] = 2; t_phase[2] = 2; t_phase[3] = 3;
cap[1] = 1; cap[2] = 1; cap[3] = 39.98;
linear_block("LIN5",spice_file,out_net,ninputs,in_nets,in_phase,t_phase,cap,fc);

/*
Interconnect the blocks in the SPICE input file
*/
printf ("\n interconnect routine :");
interconnect(spice_file);
printf ("\n input_source routine :");
/* This routine writes the input source statement in the SPICE file. */
input_source(spice_file,"IN",n1,1,0.0,1.0);
printf("\n anastat routine :");
/* This routine obtains the type of SPICE analysis and the parameters from
the user. */
anastat(spice_file);
printf ("\n output routine :");
output(spice_file,"AC",n6,2);
printf ("\n end_ckt routine :");
end_ckt(spice_file);

/*
The following statement creates pads that are connected to the circuit.
*/
Standard1("pad","pad1",n1); /* Input pad to the circuit. */
Standard1("pad","pad2",n2);
Standard1("pad","pad3",n3);
Standard1("pad","pad4",n4);
Standard1("pad","pad5",n5);
Standard1("pad","pad5",n6); /* Output pad to the circuit. */

/*
The following statements perform the analog standard cell layout of the
circuit and gets the CIF filename and the type of plot.
*/
Analog_layout(100);
printf("\n Enter name of CIF data file: ");
scanf("%s",cif_file);
printf("\n Enter a '1' for a bounding box plot or a '0' for a detailed plot: ");
scanf("%d",&plotype);

/*
Generate a CIF file from the layout information.
*/
CIF_out(cif_file,plotype);
}
End()

```

Fig. 14 - C language circuit description for the fifth-order filter of Fig. 13.

The `begin_ckt` routine writes out the title and the simulation subcircuits into the user's SPICE file. The `"linear_block"` routine called once for each instance in the user's circuit, generates a complete discrete-time subcircuit. The `"interconnect"` routine generates the linearblock subcircuit call statements

and simultaneously performs all the interconnections between the various blocks by net values. The remaining routines write out the appropriate analysis statement, I/O statement, and the end statement for the SPICE file. Fig. 15 shows the results of inputting the SPICE file generated by AIDE2 to the SPICE

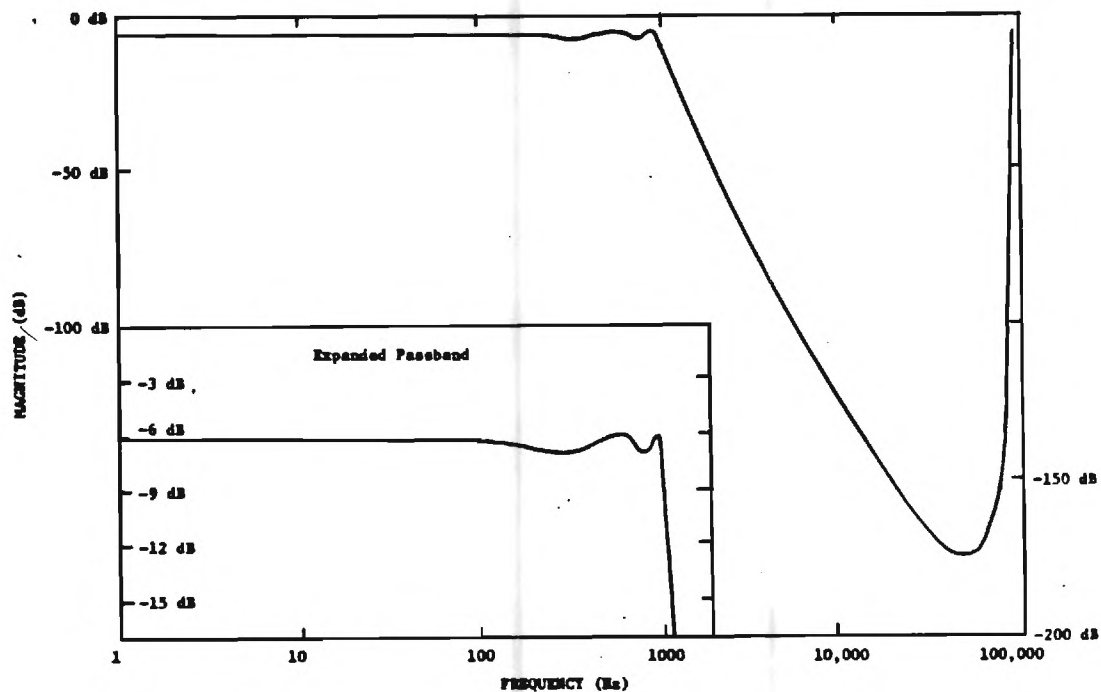


Fig. 15 - Simulation of Fig. 14 using the AIDE2 generated SPICE input file. simulation program. The AIDE2 generated SPICE input file consists of a several page listing and is not given here. Examples of AIDE2 generated SPICE input files can be found in the User's Manual [10]. It is possible to include the simulation capability within the AIDE2 program using a simulator such as SPICE-PAC [15]. This has not been done because of the problems of incorporating software developed elsewhere in a program which may be distributed to SRC industries or US universities. After the SPICE simulation is completed, then the user can direct AIDE2 to generate a CIF plot. In the execution of the CIF file, the "standard" routine assigns the cell its instance name and updates the

netlist according to the nets at its I/O port location. The "Analog_layout" routine generates a CIF file. Fig. 16 shows the bounding box layout of Fig. 13 as generated by AIDE2. Finally, the "END" statement culminates the description program. The time to define and design the filter took about 13 hours. 12 hours were spent on transforming the filter specifications to the schematic of Fig. 13 and calculating the values of the capacitors. The time to create the circuit description of Fig. 14 was approximately 30 minutes. The discrete-time frequency response simulation using the input SPICE file from AIDE2 generating

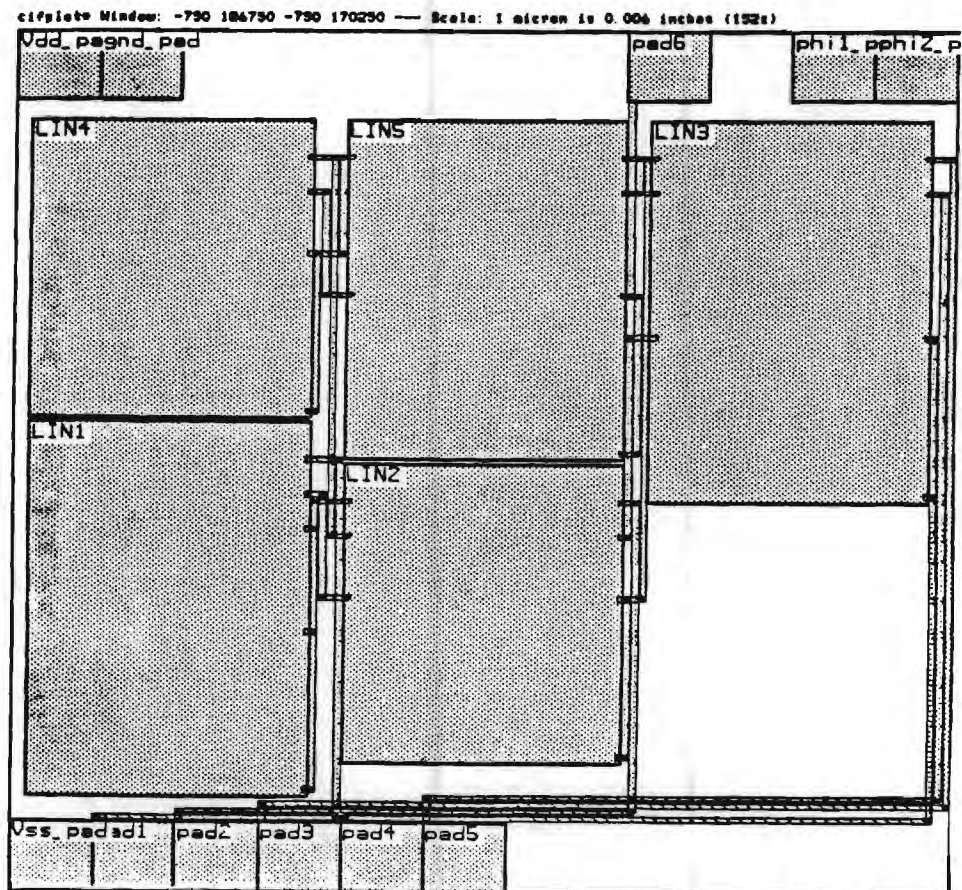


Fig. 16 - Bounding box layout of Fig. 12 generated by AIDE2.

the plot of Fig. 15 took about 5 minutes. The layout by AIDE2 took 5 minutes for the complete CIF plot. These times are for the AIDE2 program implemented on a VAX 11/780 computer using the UNIX 4.2 operating system.

7.0 USE OF AIDE2 FOR HIGH LEVEL DESIGN

Although AIDE2 is a very useful CAD tool in designing analog circuits, the above example of a 5th-order filter clearly shows that the design leverage desired by the development of analog CAD tools has not yet been achieved. The time to design, simulate, and layout the 5th-order filter without using CAD tools would probably be around 5 working days. Using AIDE2 at the intermediate level has reduced this time to approximately 1.5 working days or 13 hours. However, the next step to be taken is to reduce the time required to define and describe the design to the AIDE2 program. This should result in the ability to go from the specifications of the design to a completed layout in less than an hour. What is needed is the development of high level programs as indicated in Fig. 1. This section describes two such programs which have been written and applied to the design of analog integrated circuits. One program is designed for switched capacitor filters [5] and successive approximation analog-digital converters [16].

7.1 AROMA - An Analog Silicon Compiler for SC Filters

An area optimized CAD, switched capacitor filter design program called AROMA was developed to interface with an earlier version of AIDE2 designated as AIDE [17]. This high level CAD program contains several user-selectable filter approximation techniques. AROMA permits the user to make tradeoffs between several design parameters such as passive sensitivity, op amp output voltage swings, clock frequency, and the total capacitance of the filter. The program has default values of the tradeoffs, however, each block in the cascade filter can be modified by the user to meet specific requirements.

Fig. 17 shows a block diagram of the high level program designated as AROMA. The filter can be specified by one of the following options: 1.)

frequency specifications, 2.) continuous-time domain transfer function $H(s)$, and 3.) discrete-time domain transfer function $H(z)$. In this particular

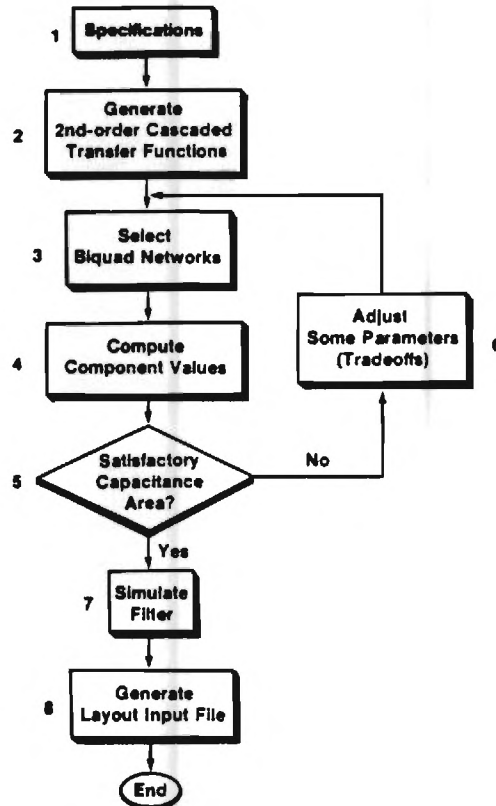


Fig. 17 - Block diagram of the high level program to design SC filters. program, the simulation is accomplished before entering the AIDE program. The output of AROMA is a circuit description program which when applied to AIDE will generate a layout.

As an example, an 8th-order, Chebyshev, bandpass filter is considered. The specifications for the filter are its Bode plot characteristics which are entered into the AROMA filter design program. AROMA calculates the capacitance values for the four cascaded biquad sections using the general SC biquad filter shown in Fig. 18. The AIDE2 circuit description for the filter is described in the C programming language as follows.

```
#include "aide2-macros"
float cap4[11] = {0,0,91,56,60,2,0,15,2,1,1}
```

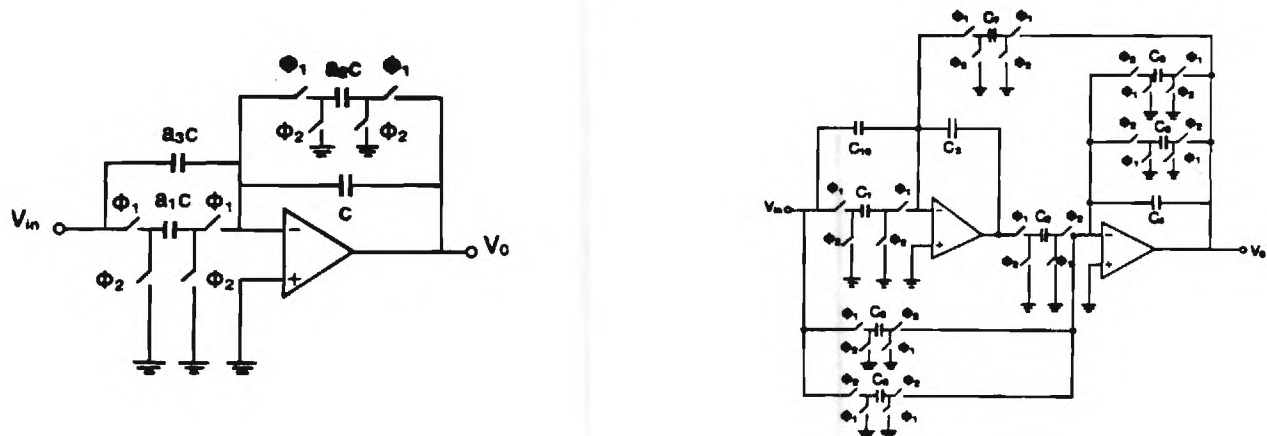


Fig. 18 - Switched capacitor biquad circuit used in AROMA.

```
float cap3[11] = {0,0,78,48,52,2,0,18,2,1,1}
float cap2[11] = {0,0,36,22,23,2,0,6,2,1,1}
float cap1[11] = {0,0,34,21,22,2,0,7,2,1,1}
main() { /*8th order SC filter*/
  Begin5("bandpass",n1,n2,n3,n4,n5) {
    BiquadSC1("Block4",n4,n5,cap4);
    BiquadSC1("Block3",n3,n4,cap3);
    BiquadSC1("Block2",n2,n3,cap2);
    BiquadSC1("Block1",n1,n2,cap1);
    Standard 1("pad","output_pad",n5);
    Standard 1("pad","input_pad",n1);}
  End() Analog_layout(100); CIF_out("F8.cif",1);}
```

BiquadSC1 is an example of how the constructs of the C language can be used to create a routine that will take the "linear_block" routine and use it to develop a higher level structure such as the biquad circuit of Fig. 18. This routine or "soft macro" greatly simplifies the circuit description as compared with the description of the fifth-order filter shown in Fig. 14. The Begin5() statement is an AIDE2 macro which expands into function calls that define the five nets, n1 to n5. The four biquads are called and pads are connected to the input and output nets. The resulting layout is shown in Fig. 19. It is 90 mils by 100 mils using a 5 micron CMOS process. For clarity, the bounding box layout option has been selected. The total capacitance for all four biquads is 610 units.

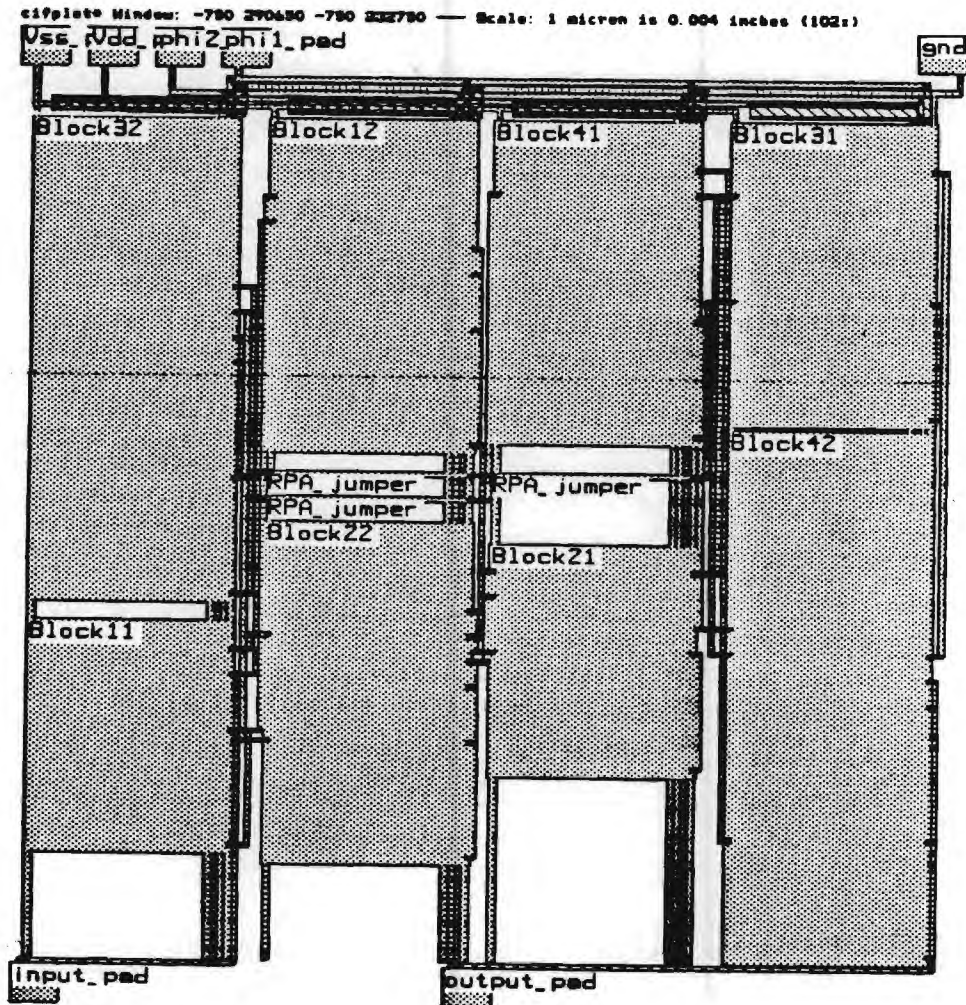


Fig. 19 - Bounding box layout of an 8th-order, switched capacitor bandpass filter.

7.2 An Analog Silicon Compiler for A-D Converters

A high level design program has been written which is capable of generating circuit description for AIDE2 of a successive approximation A/D or D/A converter has been developed [16,18]. This high level program has been

incorporated into the AIDE2 program to produce a CAD tool capable of high level input and a CIF plot output. Users with little knowledge of analog integrated circuit design can use this program to design and layout successive approximation A/D or D/A converters.

The program asks the user for the following input: 1.) the type of converter (A/D or D/A), the number of bits of resolution, and the required linearity. Based on this and information about the capacitor and resistor ratio accuracies for the particular fabrication process, the program selects the architecture which best meets the required specifications while minimizing the chip area of the converter.

The available architectures use resistors and capacitors to decode the most significant bits and the least significant bits. Which bits are decoded by resistors and which are decoded by capacitors are determined by equations describing the nonlinearity as a function of passive component mismatch.

In order to implement this program, it was necessary to design, define, and enter five new block into the AIDE2 library. These blocks include a capacitor array, a resistor string, a successive approximation register, a switch array, and a comparator. The blocks are programmable with respect to the number of bits in the converter. A description program created by the high level synthesis program makes procedural calls to the building blocks which are then placed and routed by the AIDE2 program. The simulation of the converters was not done because of the lack of an efficient simulator for a circuit containing both analog and digital circuits.

The key aspect of this high level program is the selection of the topology. The variables in this selection algorithm are whether the converter is an A/D or D/A, which bits are decoded by voltage division (resistors), which

bits are decoded by charge division (capacitors), and how many bits are decoded by each. Fig. 20 shows a flowchart of the topology selection algorithm. The

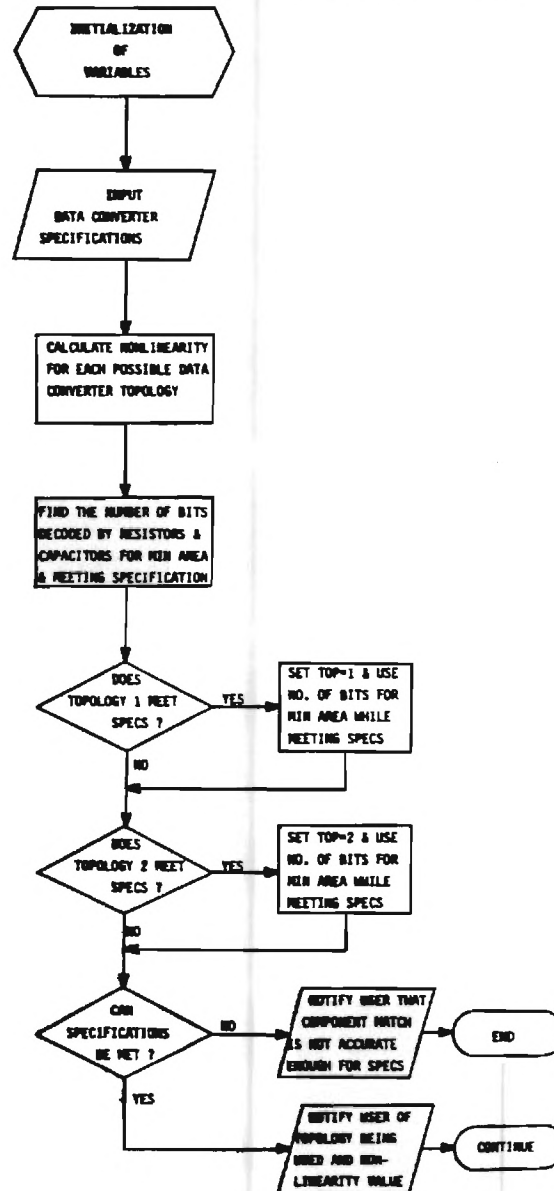


Fig. 20 - Flowchart of the topology selection algorithm.

program begins with the initialization of variables used in the program, including values for capacitor ratio accuracy, resistor ratio accuracy, the area associated with a unit capacitor, and the area associated with a unit resistor. These ratio accuracies are process dependent and should be obtained

by fabricating the actual capacitor array and resistor string on a test chip for the particular process being used.

Tables 2 and 3 tabulate the topology results from running the program for seven-bit and eight-bit data converters. Since the nonlinearities considered result from the D/A portion of the converter, the results for an A/D converter of given specifications are the same as for a D/A converter of the same specifications.

No. of Bits	Maximum Differential Linearity	Maximum Integral Linearity	Result
7	0.5LSBs	0.5LSBs	4 MSBs: Capacitors 3 LSBs: Resistors
8	0.5LSBs	0.5LSBs	Component accuracy not sufficient
8	0.5LSBs	2LSBs	Component accuracy not sufficient
8	0.5LSBs	3LSBs	4 MSBs: Resistors 4 LSBs: Capacitors
8	0.6LSBs	0.5LSBs	4 MSBs: Capacitors 4 LSBs: Resistors

Table 2 - Example topology results ($ca=47$, $ra=78$, $cm=.002$, $rm=.02$).

Table 2 shows the results for the case using the present areas of the unit capacitor and unit resistor. The area associated with a unit capacitor (ca) in the binary weighted capacitor array is 47 square units and the area associated with a unit resistor (ra) in the resistor string is 78 square units. If the resistor or capacitor is redesigned, the corresponding areas should be updated in the variable initialization portion of the description program. Table 3 shows the results for the hypothetical case of a unit capacitor area of five square units. The results of both Tables 2 and 3 were obtained using a

capacitor matching accuracy (cm) of 0.002 and a resistor ratio matching accuracy (rm) of 0.02. It can be seen that seven-bit converters with seven bits of linearity are obtainable with the matching accuracies used. For the case of eight bits, eight bits of differential linearity can be achieved at the expense of the integral linearity if resistors are used to decode the MSBs.

No. of Bits	Maximum Differential Linearity	Maximum Integral Linearity	Result
7	0.5LSBs	0.5LSBs	5 MSBs: Capacitors 2 LSBs: Resistors
8	0.5LSBs	0.5LSBs	Component accuracy not sufficient
8	0.5LSBs	2LSBs	Component accuracy not sufficient
8	0.5LSBs	3LSBs	4 MSBs: Resistors 4 LSBs: Capacitors
8	0.6LSBs	0.5LSBs	6 MSBs: Capacitors 2 LSBs: Resistors

Table 3 - Example topology results ($ca=5$, $ra=78$, $cm=.002$, $rm=.02$).

Conversely, eight bits of integral linearity can be achieved at the expense of the differential linearity if capacitors are used to decode the MSBs. It should be noted that the nonlinearity equations assume a worst case adding of the nonlinearities and the matching accuracies are statistical, so eight bits of linearity could be achieved with the accuracies used, resulting in a decreased yield.

By comparing Tables 2 and 3, one can see the effect of component area on the resulting converter topology. As the unit capacitor area is decreased, the program tends to use more capacitors and fewer resistors to achieve the required linearity. This comparison is for illustrative purposes only. In

reality, the matching accuracy would decrease as the size decreases, changing the results. The number bits decoded by capacitors and resistors is determined by the following equation for minimum area.

$$K - M = \frac{\ln(ra/ca)}{\ln 2} \quad (4)$$

where K is the number of bits decoded by capacitors and M is the number of bits decoded by the resistors. It can be seen from Table 3 that this equation is not followed for the case of eight bits of differential linearity. In this case, the program chooses a topology that sacrifices area in order to achieve the required linearity.

Table 4 shows the run times for converters of three different resolutions when a detailed layout is produced. The run time does not depend on the resolution as much as it depends on the type of converter. It is seen that D/A converters are generated in approximately only half the time it takes the

Number of bits	A/D Run Time (CPU seconds)		D/A Run Time (CPU seconds)	
	Bounding Box	Detailed Plot	Bounding Box	Detailed Plot
4	24	71	11	34
8	29	75	13	36
12	35	79	15	38

Table 4 - Example run times for bounding box and detailed layouts of D-A converters on a VAX 11/780.

program to generate A/D converters. Table 4 also shows the run times when a bounding box layout is generated. A bounding box layout is completed in less than one half the time it takes to create a detailed layout. Fig. 21 is a plot of an eight-bit A/D converter and Fig. 22 is a plot of an eight-bit D/A

converter designed with this program which interfaces with AIDE2.

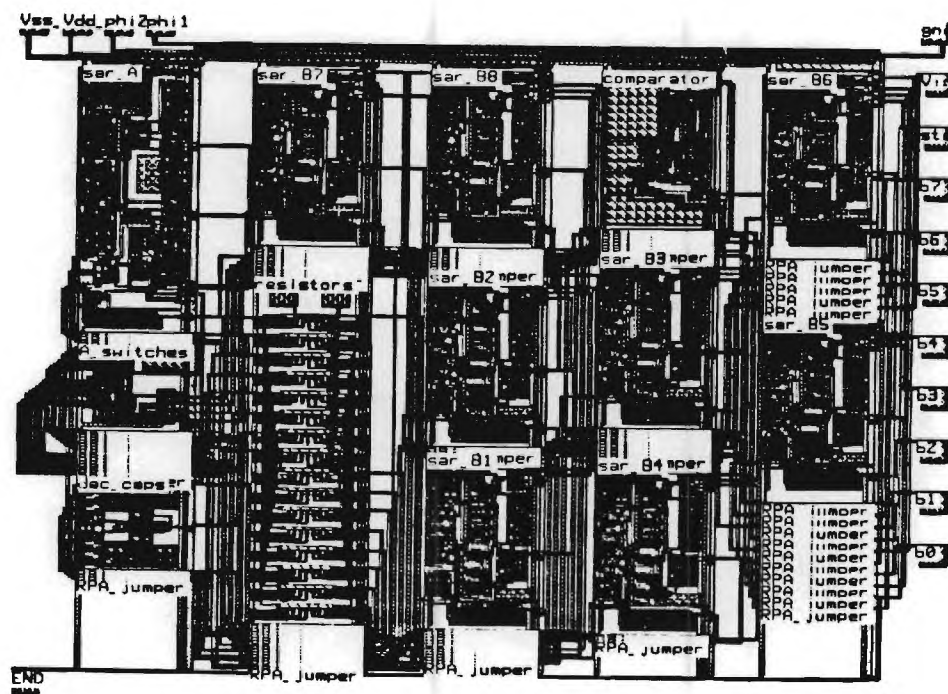


Fig. 21 - Complete geometric description of an 8-bit A/D converter.

This section describes a high level synthesis program for A/D and D/A converters developed to interface with the AIDE2 program. The result is an incremental refinement type silicon compiler for CMOS charge redistribution data converters. This technique uses programmable cells with interconnections defined by a program that translates high level specifications into a predefined circuit description. This program is useful in designing low to moderate performance data converters. Because the converters are auto-routed, they cannot achieve the high performance necessary in precision applications. The program input is the data converter specification and the program output is a file containing the geometric layout description of the circuit.

8.0 FABRICATION OF AIDE2 CIRCUITS

The AIDE2 program has been used to design approximately 17 switched

capacitor filters ranging from simple to complex in design. Fig. 23 shows one of the quadrants of the multiproject chip which is presently being fabricated

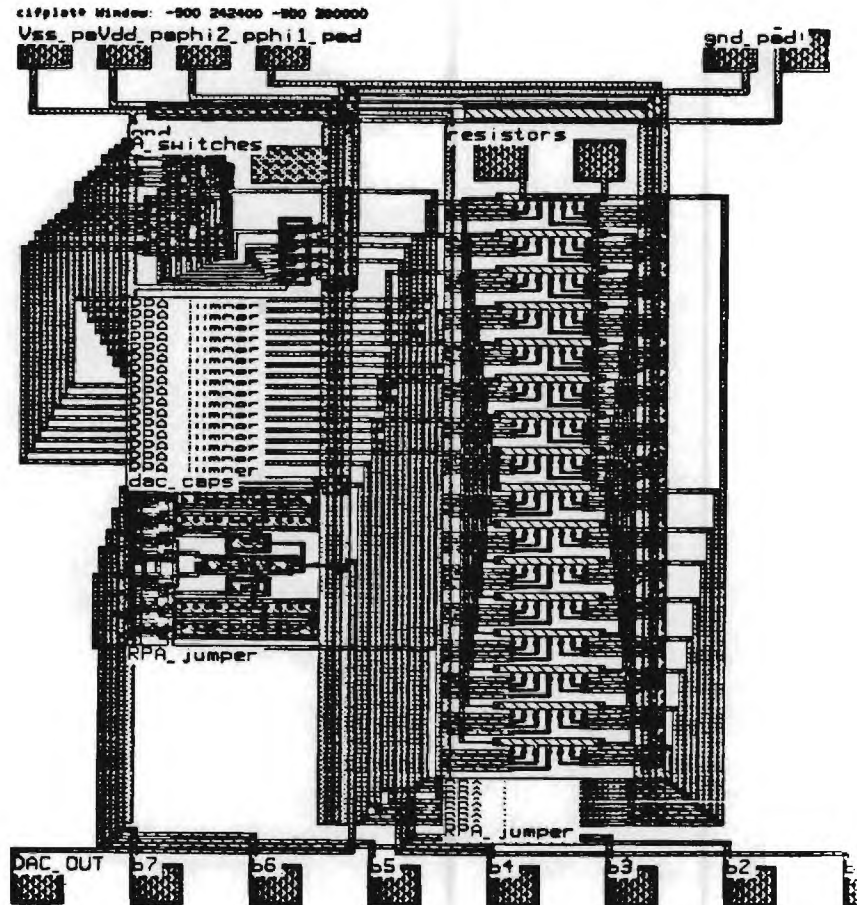


Fig. 22 - Complete geometric description of an 8-bit D/A converter.

at Harris Semiconductor Corporation using a double-poly CMOS technology described in Appendix A. One of the filters is a PCM, fifth-order filter similar to one built at Harris Semiconductor Corporation and will be used as a benchmark. The area and performance of the benchmark will be compared with the custom designed circuit. A 14th-order filter was attempted in order to benchmark a filter designed and fabricated by Silicon Systems. Unfortunately, the simulation of the filter did not match the expected response and was not fabricated. Also

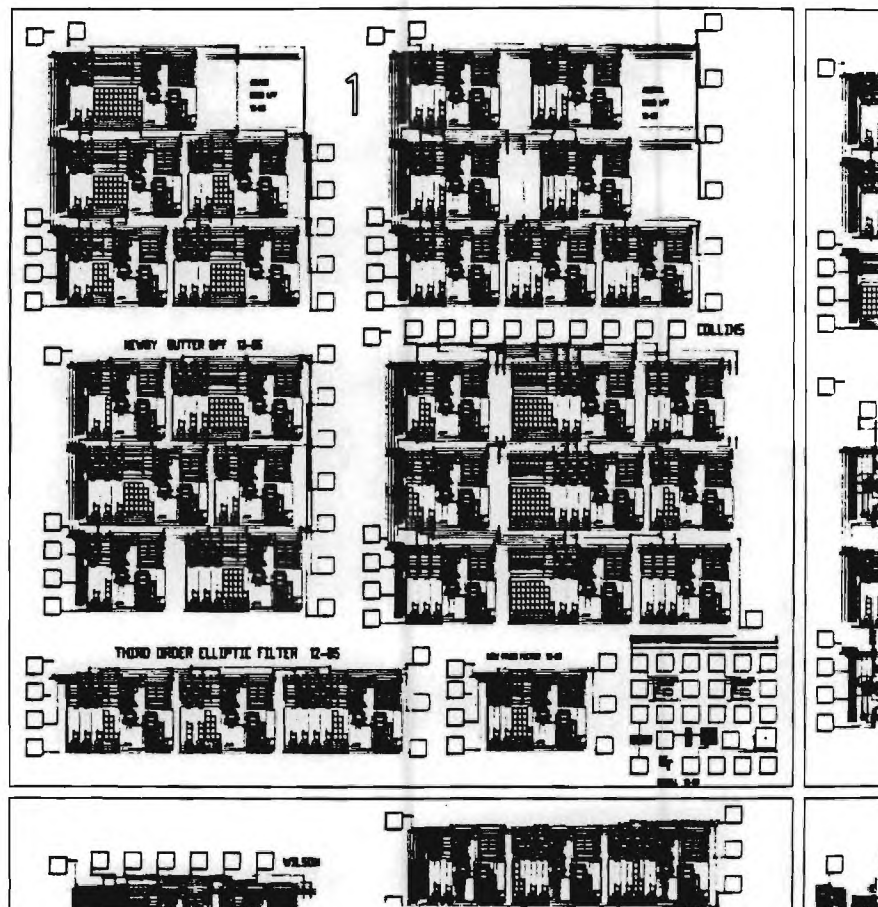


Fig. 23 - Illustration of the upper left-hand quadrant of the multiple project chip containing AIDE design circuits.

included on the multiproject chip are test circuits for evaluating the performance of a circuit such as the linear block which has been designed and layed out by a CAD program. Block performance concerns include the accuracy of the gain and time constants, clock effects, offsets, frequency limitations, dynamic range, and distortion. Unfortunately, the performance of the designs is not available at this time.

9.0 TESTABILITY OF ANALOG CIRCUITS DESIGNED BY AIDE2

The testability of analog integrated circuits is an important issue in the overall design problem. The regularity of the AIDE2 CAD program has permitted the ability to incorporate testability during the design phase. Once a system has been described to the AIDE2 program, simulation can be made of all or a part of the system. The ability to replicate experimentally any simulation has been developed and is being incorporated into the AIDE2 program. If a simulation is to be replicated, the program inserts a test cell which will allow a physical reconfiguration to achieve the simulation by entering all the test cells.

The test cell contains a single-pole, double-throw switch and a shift register. The shift register controls the position of the switch. The shift registers of all test cells are connected in series so that a desired physical reconnection can be obtained by serially loading the shift registers. The width of the test cell is 41λ which is about 20% of the height of the cell. Fig. 24 shows a block layout of a third-order filter with no testing capability. Fig. 25 shows the same filter with test circuits which will allow maximal testing. The area penalty is roughly 1.5. Fortunately, the designer would never want to maximally test a circuit (i.e. separate every block, and combinations of blocks). A more realistic area penalty is probably around 1.2. Part of the circuits being fabricated on the chip described in the last section includes a third-order filter and a maximally testable version. The objective is to determine the influence of the test circuits on the filter performance.

The next steps in this research include the definition of a fault and the isolation of this fault. Unfortunately, the definition of an analog fault is not clear. The best approach is to allow the designer to enter his own

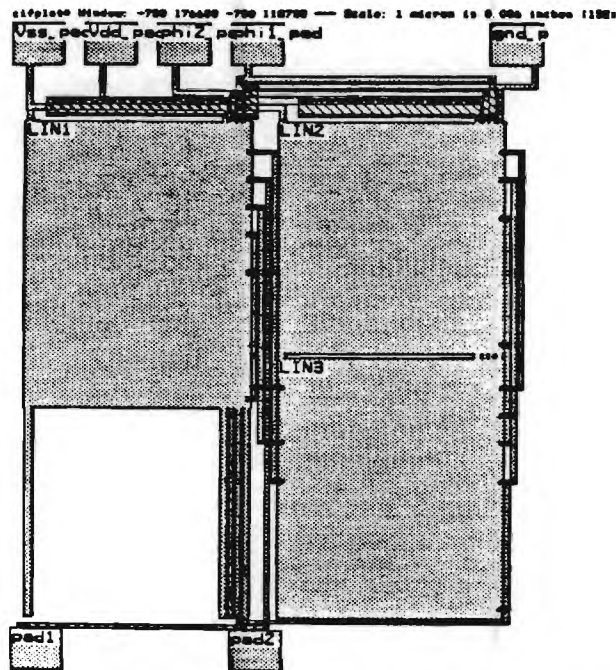


Fig. 24 - Block diagram layout of a third-order filter.

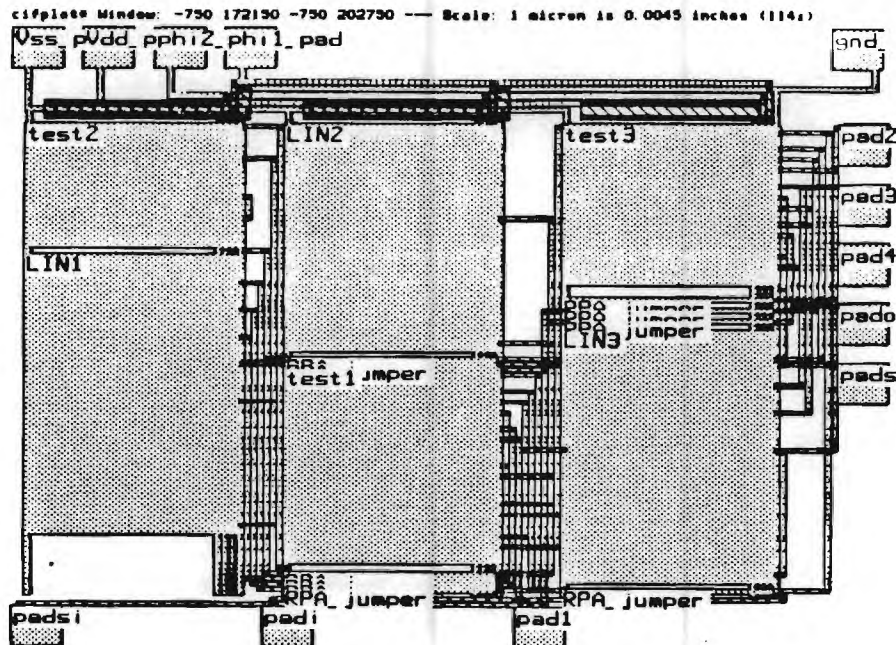


Fig. 25 - Fig. 24 with maximal testing capability.

definition that the program uses. Such a definition might be the ripple

specification at a given frequency of a filter. Once a fault has been detected, the next step is to locate the cause of the fault. The physical accessibility will be very useful in successfully achieving this objective.

10.0 TECHNOLOGY ASPECTS OF AIDE2

Originally, the objective of AIDE2 was to try to make it technology independent. The approach used was to examine several p-well CMOS technologies and to combine them into a generic, p-well CMOS technology. Unfortunately, this has not provided an optimum solution. For users without access to a technology similar to the generic technology, this can create problems. The generic CMOS process is assumed to be a p-well process having one layer of metal and two layers of polysilicon. The first poly layer is used for transistor gates, capacitor bottom plates, and interconnections within the cells and external to them. The second layer of polysilicon is used for capacitor top plates, transistor gates, interconnect within the cell, and as a shield between signal lines and clock lines. Shield is achieved by sandwiching a grounded poly 2 layer between the poly 1 layer and metal. The primary interconnecting layers is metal for horizontal (parallel to channel) and poly 1 for vertical (perpendicular to channel) routing.

The efforts to achieve a technology independent program concern two databases. These are the simulation database and the geometric database. The first step was to identify a set of parameters which define each database for the generic technology. The parameters were separated into user-defined and process-defined parameters and are illustrated in Table 5. User-defined parameters refer to those parameters in which the user has control and typically these are known during preliminary circuit design. The process-defined parameters on the other hand are dictated by the technology.

Parameter		Definition
User Defined	AS and AD	Area of the source and drain
	PS and PD	Perimeter of the source and drain
		Channel length
	W	Channel width
	T	Temperature
	Lambda	Minimum physical resolution unit
Process Defined	C_{ox} & C_{pp}	Interlayer capacitance
	N^+ , P^+ & N_{SUB}	Doping levels
	XJ	Junction depth
	RSH	Sheet resistance
	μ_N & μ_P	Surface mobility

Table 5 - Summary of user and process defined parameters.

Fig. 26 shows how the generic parameters of Table 5 are employed in the AIDE2 program. Test vehicles have been designed which when fabricated in a given technology will allow the user to determine the values of the generic parameters. These test vehicles and their background is described in more detail in the AIDE2 Users Manual [10]. This information can be converted by a program contained within AIDE2 to produce the simulation databases and geometrical database. Theoretically, each block in the AIDE2 library could call the simulation database in order to simulate the block in the particular technology. Unfortunately, this problem is more severe in the geometrical database. It is difficult to adjust the geometric description from process to process. The approach taken was to define a set of layout rules which represent the worst case of several anticipated geometries. This does not lead

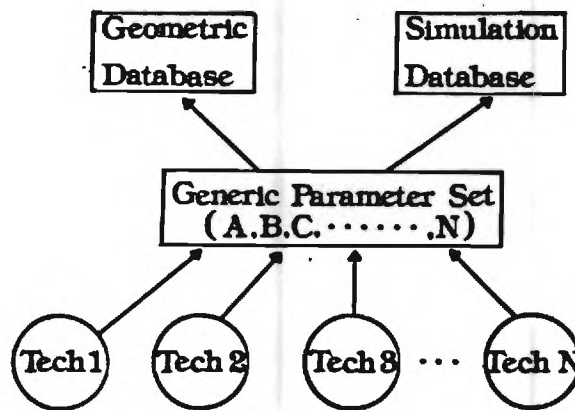


Fig. 26 - Illustration of the generic parameters used in AIDE2.

to minimum geometry configurations and aggravates the efficient use of area by AIDE2. The generic design rules used in AIDE2 are described in more detail in the AIDE2 Users Manual [10].

11.0 SUMMARY AND PLANS

This report has summarized the results of the SRC sponsored research program at Georgia Institute of Technology on computer aided design of analog integrated circuits. The major accomplishment of this program has been the development of AIDE2, a CAD program which allows the multilevel design of analog integrated circuits. This program has been debugged and used to simulate and layout switched capacitor filters. The library contains only one cell at the present and that cell is a parameterizable, first-order linear block. This cell was developed for an n-well technology. An effort was made to develop technology independence for the simulation and geometrical databases of the AIDE2 program.

Other accomplishments related to this research include the development of two high level programs to demonstrate the use of AIDE2 as a silicon compiler. One program permitted the high level design of switched capacitor filters and

the other the design of successive approximation A/D and D/A converters. Also, a method of physically isolating and connecting the various blocks layed out by AIDE2 has been accomplished. This step is the beginning of a means of insuring the testability of an analog circuit.

Another major effort during this period of time has been the implementation of the necessary hardware and software to do this research. The hardware is primarily a VAX 11/780 using the UNIX 4.2BSD operating system. The software includes the University of Washington/VLSI Consortium tools which contain the UC Berkeley software for simulation and layout. Along with effort was the development of a test facility to experimentally measure the performance of fabricated integrated circuits.

The plans for this research effort include the following. The first and most important is the evaluation of the integrated circuits which have been designed by AIDE2. There are several aspects of AIDE2 which need to be modified in order to extend its usefulness and flexibility. These aspects include the development of blocks for a generic p-well process and for the NSF/DOD sponsored quick turnaround fabrication facility [19]. Converting the cells to the NSF/DOD technology will permit fabrication without being concerned with the technology. For the present, this is probably the most expedient way of solving some of the major problems of technology independence. New blocks/cells must be added to extend the capability of the AIDE2 program. Such blocks include those for the high level analog-digital converter program described previously. Documentation of the AIDE2 program is an area which needs to be brought up to date. The present documents in development are the AIDE2 Users Manual and Programmers Manual. This documentation along with other publications should provide sufficient background for users of AIDE2.

12.0 REFERENCES

1. P.E. Allen, "A Multilevel Analog Integrated Circuit CAD Program," Research Proposal to Semiconductor Research Corporation, June 5, 1984, School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.
2. P.E. Allen, "A Computer Aided Design Methodology for LSI/VLSI," SRC Technical Report No. T85014, March 1985, Semiconductor Research Corporation, P.O. Box 12053, Research Triangle Park, NC 27709.
3. P.E. Allen, et.al., "The Use of a CAD Tool to Design Analog Integrated Circuits," Proc. of ISCAS, pp. 1223-1226, Montreal, Canada, May 1984.
4. P.E. Allen, E.R. Macaluso, S.F. Bily, and A.P. Nedungadi, "AIDE2: An Automated Analog IC Design System," Proc. of IEEE 1985 Custom IC Conf., pp. 498-501, Portland, OR, May 1985.
5. E. Sanchez-Sinencio and J. Ramirez-Angulo, "AROMA: An Area Optimized CAD Program for Cascade SC Filter Design," IEEE Trans. on CAD of IC's, Vol. CAD-4, No. 3, pp. 285-303, July 1985.
6. E.R. Macaluso, "A Procedural Language for Automated Analog Circuit Design," MS Thesis, Dept. of Electrical Engineering, Texas A&M University, Dec. 1984.
7. S.F. Bily, "A Functional Cell Design Methodology for Analog Integrated Circuits," MS Thesis, Dept. of Electrical Engineering, Texas A&M University, May 1985.
8. R.W. Hon and C.H. Sequin, "A Guide to LSI Implementation," 2nd Edition, Xerox Palo Alto Research Center, Palo Alto, CA Jan. 1980.
9. L.W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," Electronics Research Laboratory, Memorandum No. ERL-520, College of Engineering, University of Calif., Berkeley, CA, May 9, 1975.
10. P.E. Allen, R.L. Asibal, E.R. Macaluso, and D.L. Ellsworth, "AIDE2 Users Manual," School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, April 1986.

11. P.E. Allen and P.R. Barton, "Development of a Synthesis Program for Successive Approximation Data Converters," Proc. of 28th Midwest Circuits and Systems Symposium, pp. 305-307, Louisville, KY, August 1985.
12. P.E. Allen and D.L. Ellsworth, "AIDE2 Programmers Manual," School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, April 1986.
13. B.D. Nelin, "Analysis of Switched-Capacitor Networks Using General-Purpose Circuit Simulation Programs," IEEE Trans. on Circuits and Systems, Vol. CAS-30, No. 1, pp. 43-48, Jan. 1983.
14. A. Hashimoto and J. Stevens, "Wire Routing by Optimizing Channel Assignments," 8th Design Automation Workshop Proc., pp. 214-224, 1971.
15. W.M. Zuberek, "SPICE-PAC:2G6a.84.05 User's Guide", Dept. of Computer Science, Memorial University of Newfoundland, St. John's, NFLD, Canada A1C 5S7, May 1984.
16. P.E. Allen and P.R. Barton, "A Silicon Compiler for Successive Approximation A/D and D/A Converters," Proc. of IEEE CICC, Rochester, NY, May 1986.
17. P.E. Allen, "A Computer Aided Design Methodology for Analog LSI/VLSI," SRC Technical Report No. T85014, March 1985, Semiconductor Research Corp., P.O. Box 12053, Research Triangle Park, NC 22709.
18. P.R. Barton, "A Synthesis Program for CMOS Successive Approximation A/D and D/A Converters", MS Thesis, Feb. 1986, School of Elect. Engr., Georgia Institute of Technology, Atlanta, GA 30332.
19. D. Cohen and V.C. Tyree, "Quality Control From the Silicon Broker's Perspective," VLSI Design, July/August 1982, pp. 24-30.

SRC REPORT/PUBLICATION SUBMISSION FORM

PLEASE COMPLETE THIS FORM FOR ALL REPORTS AND PUBLICATIONS SUBMITTED TO SRC

PRODUCED UNDER SRC CONTRACT? Yes/No/ <u>In Part</u> SRC CONTRACT # <u>84-07-051</u>				
OTHER SOURCES OF SUPPORT? <u>Schlumberger Chair Funds and GT E-Funds</u>				
NAME OF PERSON RESPONSIBLE		NAME AND ADDRESS OF ORGANIZATION		
Phillip E. Allen Schlumber Professor		School of Electrical Engineering Georgia Institute of Technology Atlanta, GA 30332		
REPORT TITLE				
Computer Aided Design and Modeling of Analog Integrated Circuits				
REPORT AUTHOR(S)				
Phillip E. Allen				
TYPE OF REPORT (check one)	PERIOD COVERED (if applicable)	REPORT DATE	NUMBER OF PAGES	DATE SUBMITTED TO SRC
<input type="checkbox"/> Quarterly Progress	FROM <u>01/01/86</u>	<u>02/12/87</u>	<u>37</u>	
<input checked="" type="checkbox"/> Annual Report				
<input type="checkbox"/> Final Report	TO <u>12/31/86</u>			
<input type="checkbox"/> Thesis				
<input type="checkbox"/> Publication Preprint to be submitted to _____ date _____				
<input type="checkbox"/> Presentation to be presented at _____ date _____				
<input type="checkbox"/> Other (please describe) _____				
ABSTRACT OF CONTENTS OF THIS SUBMISSION (not to exceed one single spaced page)				
See attached.				
SUBJECT KEYWORDS (circle keywords supplied on reverse - note any additions)				
LIST A: Analog, CAD, CMOS LIST: B: IC/CAD Tools, auto. layout, dev. modeling				
*** FOR SRC USE ONLY ***				
DATE RECEIVED _____		SRC PUBLICATION ID _____		

COMPUTER AIDED DESIGN AND MODELING OF ANALOG INTEGRATED CIRCUITS

Abstract

This report describes the progress made in the development of CAD design and modeling methodology for analog IC's. The AIDE2 program has been used to design analog IC's which were fabricated but did not work because of two fatal layout errors. A second fabrication run has been initiated. New cells have been entered into AIDE2 which allow it to design analog-digital converters. The technology is being converted to the MOSIS 3 micron CMOS technology which is widely available to designers. The AIDE2 program has also been used to implement two applications specific analog silicon compilers.

The accuracy of the small signal models of present simulators is very poor. A method of improving the accuracy has been developed and applied to both short and long channel CMOS devices for two different technologies. The methods use a combination of table look-up and analytical modeling techniques. The accuracy of the model improves as the number of points in the table increases. The model uses the dc solution of existing large signal models as the input. Accuracy as good as the dc solution has been achieved.

The third major effort of this research involves the development of a multilevel simulator for analog circuits. A high level simulator using graphical solution methods has been combined with a low level simulator (SPICE) to permit the simultaneous simulation of an analog circuit using different simulation primitives. Typically, the high level portion of the analog circuit is linear while the low level is nonlinear.

This research in analog CAD has developed the methodology to shorten the design time and increase the probability of success for analog integrated circuits. This capability will permit the system designer to utilize existing and new VLSI technologies.

ANNUAL PROJECT REPORT

SRC Contract No. 84-07-051

February 12, 1987

COMPUTER AIDED DESIGN AND MODELING OF ANALOG INTEGRATED CIRCUITS

Prepared for

Dr. Ralph K. Cavin, III

Manager, Design Sciences Program

Semiconductor Research Corporation

300 Park Drive

P.O. Box 12053

Research Triangle Park, NC 22709

by

Phillip E. Allen

School of Electrical Engineering

Georgia Institute of Technology

Atlanta, GA 30332

COMPUTER AIDED DESIGN AND MODELING OF ANALOG INTEGRATED CIRCUITS

Abstract

This report describes the progress made in the development of CAD design and modeling methodology for analog IC's. The AIDE2 program has been used to design analog IC's which were fabricated but did not work because of two fatal layout errors. A second fabrication run has been initiated. New cells have been entered into AIDE2 which allow it to design analog-digital converters. The technology is being converted to the MOSIS 3 micron CMOS technology which is widely available to designers. The AIDE2 program has also been used to implement two applications specific analog silicon compilers.

The accuracy of the small signal models of present simulators is very poor. A method of improving the accuracy has been developed and applied to both short and long channel CMOS devices for two different technologies. The methods use a combination of table look-up and analytical modeling techniques. The accuracy of the model improves as the number of points in the table increases. The model uses the dc solution of existing large signal models as the input. Accuracy as good as the dc solution has been achieved.

The third major effort of this research involves the development of a multilevel simulator for analog circuits. A high level simulator using graphical solution methods has been combined with a low level simulator (SPICE) to permit the simultaneous simulation of an analog circuit using different simulation primitives. Typically, the high level portion of the analog circuit is linear while the low level is nonlinear.

This research in analog CAD has developed the methodology to shorten the design time and increase the probability of success for analog integrated circuits. This capability will permit the system designer to utilize existing and new VLSI technologies.

TABLE OF CONTENTS

1.0	Introduction.....	1
2.0	AIDE2: An Analog CAD Platform.....	3
2.1	Review of AIDE2.....	3
2.2	Measurements on the Harris MPC1 Chip.....	5
2.3	Modifications and Improvements in AIDE2.....	10
2.4	Technological Dependence and New Cells for AIDE2.....	11
2.5	Application of AIDE2 as a Silicon Compiler.....	14
2.6	Future Plans for AIDE2.....	17
3.0	Precision Analog Small Signal Modeling.....	18
4.0	Multilevel Simulation of Analog Circuits and Systems.....	25
5.0	Summary and Plans.....	31
5.1	Summary.....	31
5.2	Plans.....	34
6.0	References.....	36

1.0 INTRODUCTION

The general goals of the Design Sciences program of the Semiconductor Research Corporation (SRC) are focused on the design methods, test methods, and chip architectures that will support the chip performance, functionality, and producibility requirements of systems to be designed 8 years from now. The pertinent SRC goals to this research are to be able to "design systems that enable the design of chips with the specified complexity using less than 6 engineering man-months of effort beginning with logic (or high level) entry." [1].

The objective of this research in the area of computer aided design and modeling of analog integrated circuits is to develop methodologies for analog circuit design which can be implemented by the computer. This research supports the SRC goals by developing methodologies for analog integrated circuit and implementing them in a software platform that can be used together with digital CAD tools to design and layout signal processing systems containing both analog and digital circuitry.

This report describes the results of the third year (1985-86) of research in the area of analog CAD methodology and its application to analog integrated circuit design. The results of the previous two years have been described in earlier reports [2,3]. The primary result of this effort to date is the development of a CAD platform designated as AIDE2 [4] which can be used to design analog integrated circuits at multilevels and can be used as a silicon compiler for specific applications [5,6].

The development of AIDE2 and its application to design analog integrated circuits has resulted in the identification of several important problems that have been examined during the period of this report. These problems include the lack of a precise small signal model for short channel devices and the

requirement for analog simulation at higher levels than the component level. The solution of these problems should help to provide the tools and methodology by which the goals of the SRC Design Science Program can meet their objectives for analog circuits and systems.

This report will review the accomplishments and status of the AIDE2 software. The results of the program to layout integrated circuits which have been fabricated will be described. Modifications and improvements to AIDE2 will be discussed. The results of using AIDE2 to achieve an applications-specific silicon compiler will be given.

This report will also examine the progress of the research on precision analog small signal models. The various approaches possible will be reviewed along with the approach selected and why. The development of the techniques and algorithms necessary to implement the model will be discussed. The results of the model for channel conductance will be presented for two different CMOS technologies for channel lengths down to 1.2 microns. The development of two circuits which will be used to benchmark the model are given.

Thirdly, the report will describe the progress of a multilevel simulator for analog circuits and systems. The problem will be examined from a hierarchical viewpoint and the various approaches considered. A high level analog simulator will be described along with its application to the solution of a multilevel analog systems. Plans and directions for this research effort will be discussed.

Finally, this report will briefly examine some of the analog CAD resources that have been developed as a result of this research effort. These resources include software, hardware, publications, and graduate students skilled in the area of analog circuit design and analog CAD methodology.

2.0 AIDE2: An Analog CAD Platform

This section will review the AIDE2 program and its objectives. The results of the AIDE2-designed integrated circuits fabricated by Harris Semiconductor will be described along with the problems which resulted. New modifications to and improvements of AIDE2 include a partial simulation capability, an automated test capability, and new cells for the MOSIS technology [7]. This section will also examine the use of AIDE2 as a silicon compiler for application-specific analog circuits and systems. The plans for continued research with AIDE2 will be discussed.

2.1 Review of AIDE2

The AIDE2 program is the result of three years of development and evolved from a prototype version designated as AIDE [8]. Fig. 1 illustrates the inputs and outputs of the program. The output of the program is a simulation input file and a complete layout. The program has been used to design switched capacitor circuits at both the circuit level [9] and systems level [6]. Fig. 1 shows how the AIDE2 program is organized. In the AIDE2 program, C language software functions are available for defining the two main parts of the description and the chip layout. The circuit description part is specified using unique subroutine calls to particular parameterized and standard cells that are stored in the cell library facility of AIDE2. The hierarchical organization of the circuit description is such that the lowest level calls point to fixed or parameterized cells and the arguments of these calls simply define the netlist information and the cell's characteristics. This arrangement of calling into service certain combinations of parameterized and standard cells is quite useful in the synthesis of various configurations of system level analog circuits.

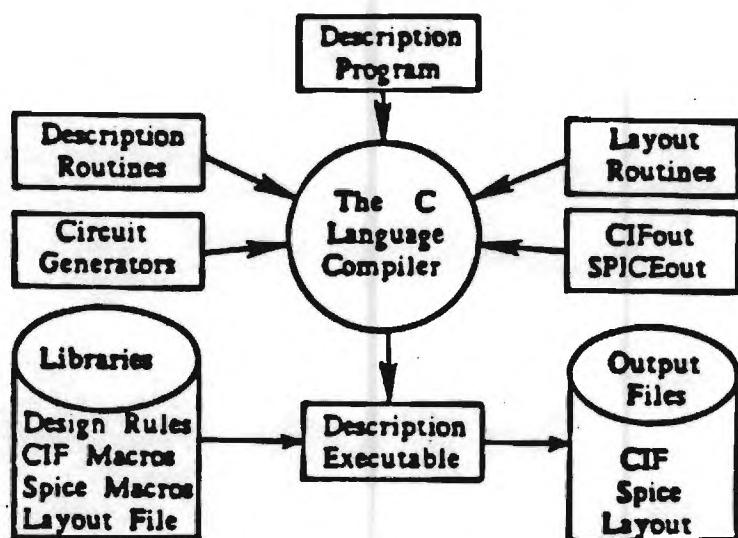


Fig. 1 - AIDE2 organization.

AIDE2 uses a procedural circuit design language in which circuits are described by procedural calls in a C language program. This program, referred to as a description program, consists of 2 parts, the circuit description and the layout control which are executed sequentially. The circuit description section specifies the netlist of the circuit. Each C language subroutine represents a cell or a macro of cells. The layout control section calls subroutines which perform a standard cell layout. At the end of the layout, the CIF_out routine is called to output the circuit geometries in the CalTech intermediate format (CIF) [10]. As the circuit description executes, the input simulation files for SPICE [11] are generated.

The description program may be written as a single circuit description or as an interactive design system which generates a circuit description resulting in circuit synthesis. The circuit layout procedure is in the form of a standard cell layout in which I/O ports are automatically connected to pads for bonding or brought out to the edge of the layout area.

AIDE2 is written in the C language and is implemented on a VAX 11/780 computer using the UNIX 4.2 operating system. It requires the software

necessary to plot and scale a CIF description of a circuit as well as the SPICE computer program which is necessary to simulate the circuit. Other supplementary software such as KIC2 can be used to view the output plot. All the necessary software is available from the University of Washington/Northwest VLSI Consortium, Dept. of Computer Science, FR-35, University of Washington, Seattle, WA 98195.

2.2 Measurements on the Harris MPC1

In the Fall of 1985, AIDE2 was used to design 17 switched capacitor filters ranging from very simple to complex. Fig. 2 shows the computer plot

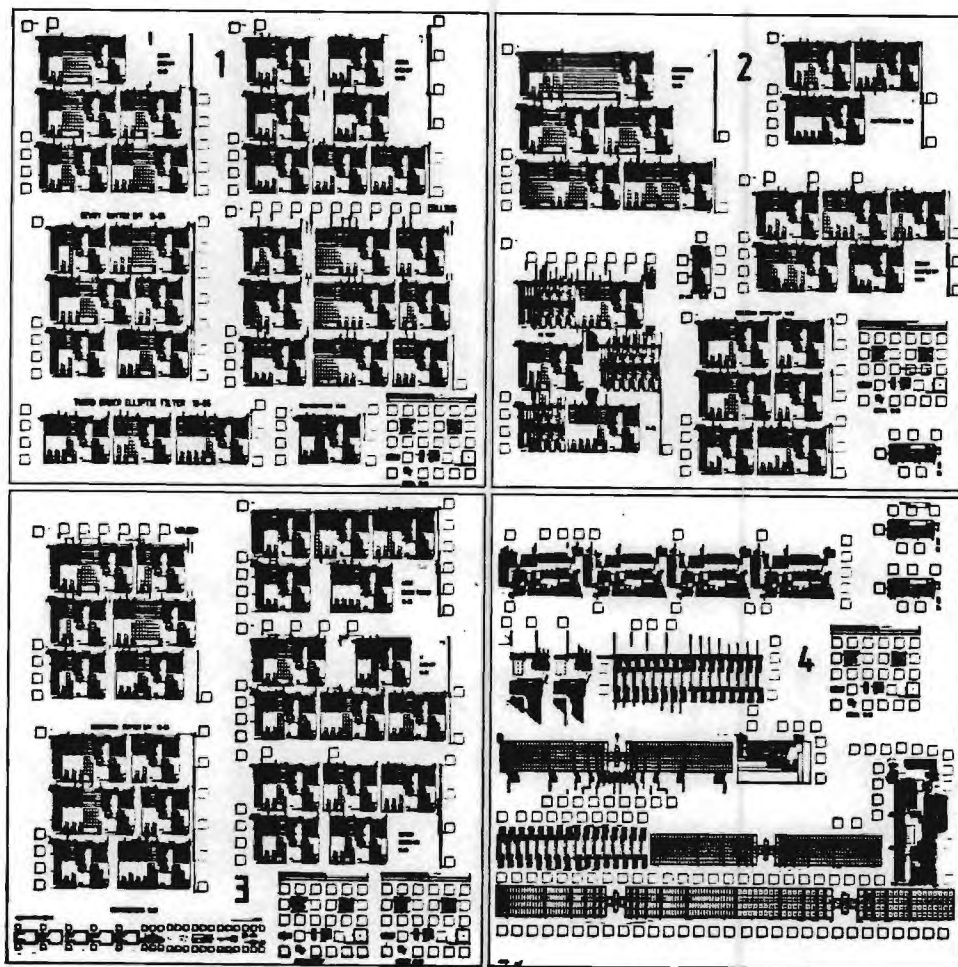


Fig. 2 - Computer layout of the Harris Multiple Project Chip No. 1.

of the multiproject chip which was fabricated at Harris Semiconductor Corporation using a 5 micron, double-poly, CMOS technology. Most of the AIDE2 designed circuits are located in the quadrants labelled 1, 2, and 3. All of the circuits designed by AIDE2 were done at the intermediate level where the designer writes a C language description program describing the design to AIDE2.

Fig. 3 shows a photograph of one of the fifth-order, low-pass filter which was designed as a benchmark filter (this circuit is located in the upper, right-hand corner of quadrant 1 of Fig. 2). Also included on the multiproject chip were test circuits for evaluating the performance of the linear block which is a key element in the CAD program. The concern of the block performance includes the accuracy of the gain and time constants, clock effects, offsets, frequency limitations, dynamic range, and distortion.

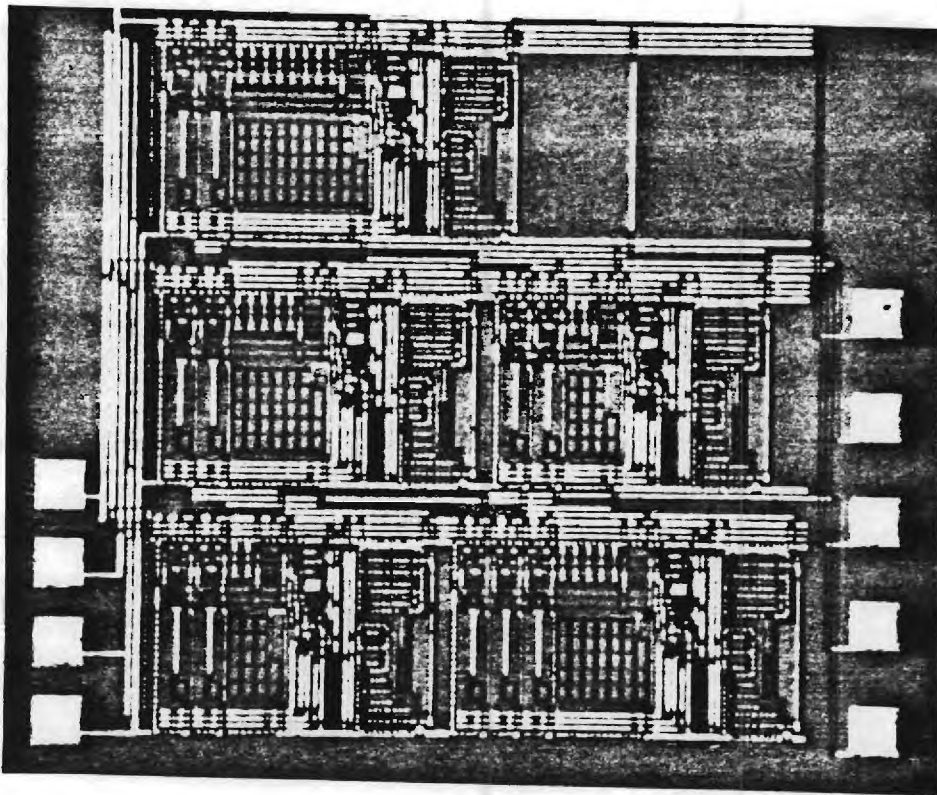


Fig. 3 - Photograph of a fifth-order, low-pass, benchmark filter.

In May, the integrated circuits designed by AIDE2 and fabricated by Harris Semiconductor were returned for testing and evaluation. The first step was to measure the device and component performance. Table 1 summarizes the comparison of these measurements with the specified electrical parameters.

Electrical Parameter	Measured Value			Specified Value	Units
	Low	Medium	High		
Threshold voltage					
NMOS	0.70	0.71	0.72	0.7	Volts
PMOS	-0.65	-0.65	-0.65	-0.7	Volts
Transconductance					
NMOS	20.03	28.3	28.85	35	Volts/ μA^2
PMOS	9.87	10	10.33	12	Volts/ μA^2
Channel modulation					
NMOS	0.0069	0.0077	0.0115	Not given	Volts ⁻¹
PMOS		0.0121		Not given	Volts ⁻¹
BJT Beta at 4mA		200		Not given	A/A
Poly I Sheet Resistance	15.06	15.46		20	Ohms/Sq.
Poly II Sheet Resistance		73.4		100	Ohms/Sq.
P+ Moat		24.44		Not given	Ohms/sq.
P+ Diffusion				40	Ohms/sq.
p Well		27.056		Not given	Ohms/sq.

Table 1 - Comparison of electrical process parameters for MPC1.

Next, the op amp used in the linear block was measured. The op amp schematic is shown in Fig. 4. The op amp is a simple, two-stage op amp using Miller compensation [12]. The op amp was originally designed for an n-well, CMOS process and was converted to a p-well process simply by inverting power supplies. It performed satisfactorily and the comparison between simulated results and experimental measurements is summarized in Table 2.

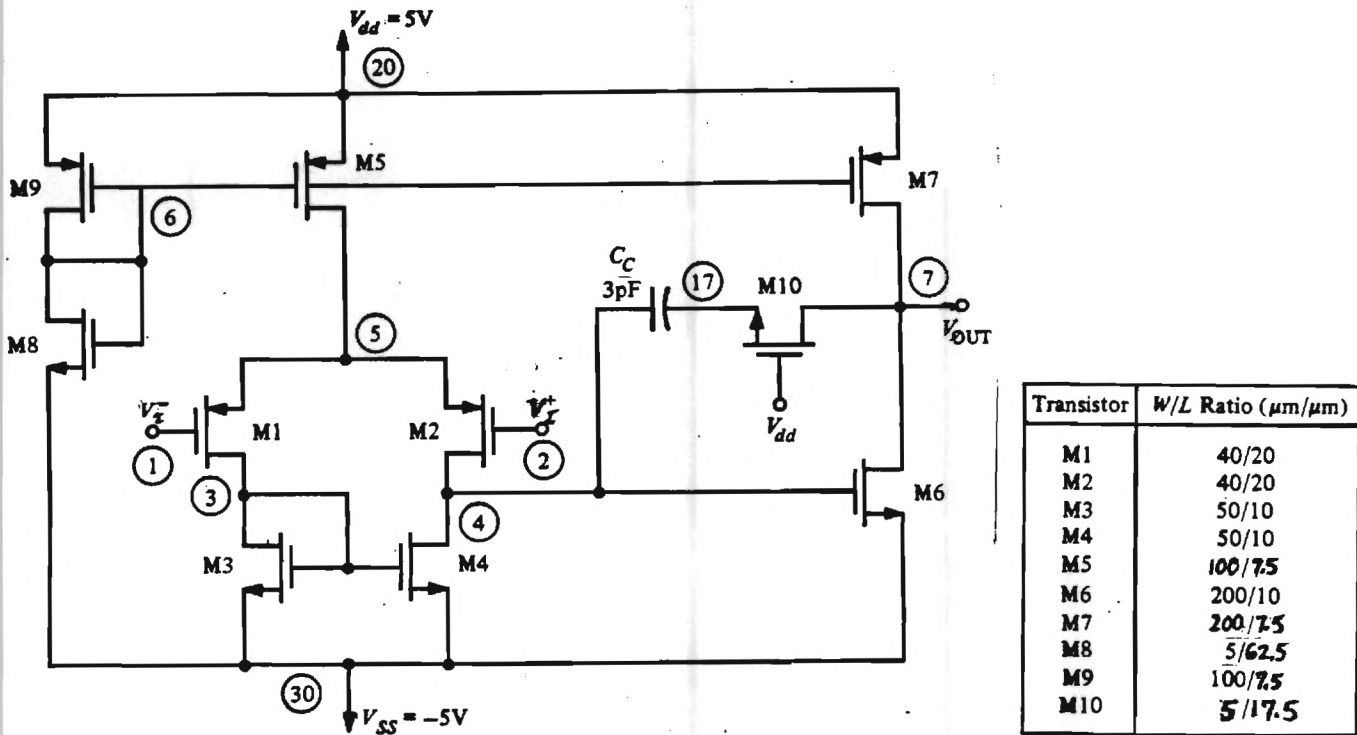


Fig. 4 - Schematic and W/L ratios for AIDE2 op amp.

Op Amp Parameter	Simulated	Measured	Units
Voltage Gain	5,884	4,615	V/V
Gainbandwidth	3.766		MHz
Phase margin (CL = 20pF)	68		Degrees
Slew rate (CL = 20pF)	+27.3/-6.8		Volts/ μs
Output Voltage Swing		-5 to 4.7	Volts
Input Common Mode Range	+4.6 to -4.4	4.3 to 4.7	Volts
Power Dissipation	2.827		mW

Table 2 - Summary of op amp performance at +/-5 Volts and 27°C.

Initial test results on the AIDE2-designed filters showed that none of the circuits worked at all. The output of each op amp appeared to be latched to power supply. A visual inspection of the op amp showed that the inputs to the op amp were reversed. This resulted in positive feedback which was the cause of the latch-up problem. This problem was corrected by microsurgery techniques and testing continued.

After the op amp connections were corrected, the filters still did not perform correctly. Another close visual inspection revealed a layout error in the transfer slice of the linear building block. In the transfer slice, two inverters are used to obtain the inverse of ϕ_1 and ϕ_2 . It was found that one end of each inverter was not connected to the appropriate power supply causing the inverters to function correctly only on half of the clock cycle. Fig. 5 is a photograph of the two layout errors.

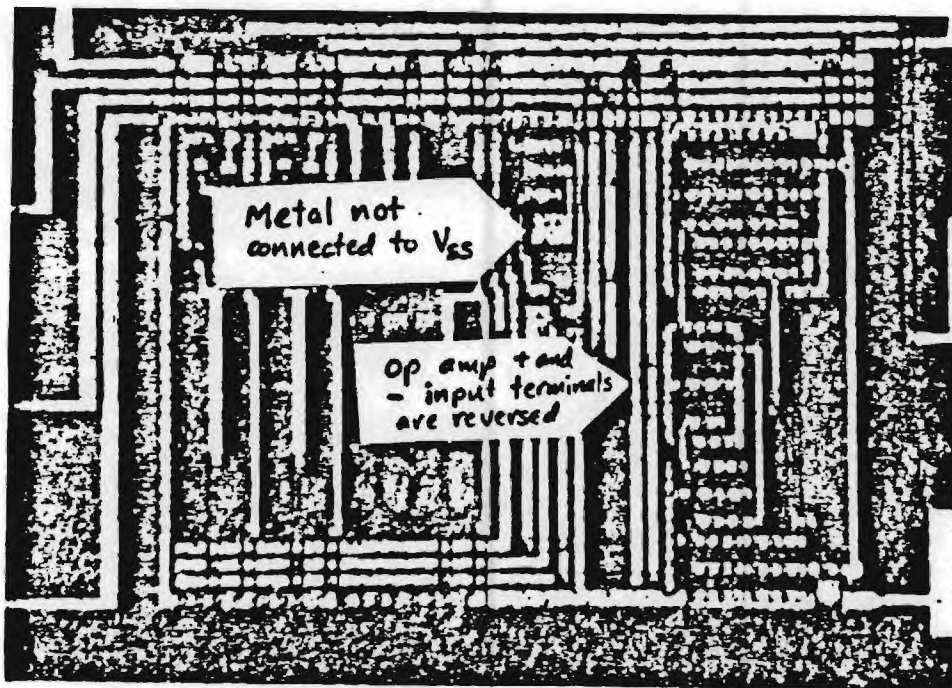


Fig. 5 - Photograph of the first-order linear block illustrating the errors found in testing.

Unfortunately, microsurgery techniques could not be used to correct the second problem. As a result, the AIDE2 circuits designed on Harris MPC1 could not be evaluated further. It is no small amount of embarrassment that two simple layout errors prevented the results from being evaluated. The above errors have been corrected and the AIDE2 used once more to create switched capacitor filters and are presently being fabricated on the second Harris Semiconductor project called MPC2. The circuits in fabrication have been carefully extracted both by hand and by computer to make sure that simple layout errors do not occur again.

2.3 Modifications and Improvements in AIDE2

In addition to correcting the above layout problems, several other bugs concerning the pad spacing and routing have been identified and removed. Other changes for purposes of convenience such as increasing the number of arguments in subroutine calls have been made. In June, the AIDE2.1 User's Guide containing approximately 170 pages of information on the use of AIDE2 was completed. A Programmer's Guide to AIDE2 is under development.

Originally, the simulation capability of AIDE2 was restricted to the input and output terminals/ports of a system. The ability to access internal nodes and to separate parts of the circuit have been introduced. This has been accomplished by introducing a voltage-controlled voltage-source at the appropriate place in the simulation file of the circuit to effect the desired topological changes. If the node is connected, the gain of the controlled source is unity. Otherwise, the gain is zero if the node is disconnected.

The partial simulation capability has been used to implement a test methodology which allows the experimental replication of any partial simulation [13]. The structured physical regularity of the AIDE2 layout provides a means of being able to physically isolate/access any port or

combination of ports of a system. If the designer desires to experimentally replicate the partial simulation, the program will automatically place and route a test cell for each node which is disconnected or accessed externally.

The test cell consists of a switch controlled by a shift register which is part of the test cell. The original port/node of the analog cell is routed to the test cell. The test cell then either connects it to its original destination or to an external pad depending upon the contents of the shift register. Only those ports/nodes which are to be disconnected and/or externally accessed need a test cell. All shift registers of all test cells are serially connected in order to provide control of the externally accessible test ports. When the designer makes a partial simulation involving internal nodes, the AIDE2 program provides the option of placing and routing the necessary test cells. A third-order filter with the degree of accessing internal nodes and isolating blocks varying from none to maximum has shown an area penalty of 1.76 required for the worst case testability conditions.

2.4 Technological Dependence and New Cells for AIDE2

The only cells in the AIDE2 library at the time of the MPC1 fabrication was a linear block capable of designing switched capacitor filters, a pad cell and an RPA jumper used for routing across cells. The linear block is programmable with respect to the topology and value of capacitors. The other two cells are fixed. Unfortunately, the linear block was originally designed for a Texas Instruments, 5 micron, n-well, double poly, CMOS process. Since the Harris MPC1 process is p-well this created some problems. With care, the power supplies could be inverted and the circuits would still work. However, this solution fails if any cells are used that were designed for a p-well process. Thus it became necessary to redesign the linear block for a p-well process. In addition, 6 new cells based on a p-well technology have been

added to the library to implement the design of successive approximation A/D and D/A converters. These cells include a comparator, switches, successive approximation registers (SAR), resistors, and capacitors. The resistor and capacitor cells are programmable while the other cells are fixed.

One of the original objectives in developing AIDE2 was to achieve technological independence. This means that with little effort, cells designed for one technology would work for a different technology. The technologies were restricted to 3 to 5 micron, double-poly, CMOS processes. There are two problems which must be solved before this objective can be realized. The first is to be able to have the simulation files adapt to the change in electrical parameters that would occur when the technology is changed. The second is to be able to have the layout files adapt to a change in layer definition or design rules when technology changes.

The approach to solving these two problems and the results were not successful. A test cell was developed that when fabricated in the desired technology would provide the electrical parameters. The next step which was not implemented would be for the simulation files to read these parameters for the passive and active model information. Unfortunately it was found very difficult to design cells that would still work for the spread of electrical parameters that existed among three CMOS technologies that were investigated.

In order to solve the geometry problem, a set of generic layers and generic design rules were developed [4]. The generic layers are illustrated in Table 3. In many cases the various layers are the same levels. This scheme worked well until one process was found to have a layer defined as the interaction of two different layers. While this problem can be solved, it was found that the only way to solve the differing design rules was to take the worst case of all anticipated technologies resulting in poor area usage [4].

LAYER	CODE	COLOR
x-well	CXW	Brown
x-moat	CXM	Orange
xb-moat	CXBM	Green
x-gate	CXG	Red
xb-gate	CXBG	Red
Capacitor bottom plate	CCBP	Red
Capacitor top plate	CCTP	Purple
Resistor	CR	Red
Metal	CM	Blue
Contact	CC	Black
Pad-opening	CPO	Black
Mask	CMS	Black
x+ guard ring	CXGR	Sky blue

Table 3 - CIF Layers and color codes for generic x-well process.

Rather than spend the time necessary to solve this problem, it was decided to designate a specific technology. The technology chosen was the 3 micron, double poly, CMOS process provided by the silicon broker designated as MOSIS [14]. This would allow general usage of the AIDE2 program in an evaluation mode to determine whether or not it was worth the effort to change the simulation and geometrical databases to a different technology. At the present time, a linear block similar to the one used for the Harris Semiconductor fabrication has been designed and entered as one of the library cells in AIDE2. A third-order and fifth-order switched capacitor filter has been designed and will be submitted for fabrication on the next MOSIS 3-micron, double poly fabrication cycle. The status of the cells in the AIDE2 library is shown in Table 4.

An alternate approach to solving the technology dependence problem is being considered using the concept of a cell compiler. The cell compiler would provide a rapid means of entering new cells which are designed for a specific technology. Rather than achieving technology independence, a user could quickly develop the desired cells for a given technology thus enabling

AIDE2 to be customized for a given user. This approach has the advantage of minimum area, rapid entry of new cells including the software to program parameterized cells, and user customization.

Cell Name	Technology			Parameterized (P) or Fixed Geometry (FG)	Name of Geometric CIF File
	p-well	n-well	MOSIS		
Comparator	x			FG	comp.cif
Switches	x			FG	swtch.cif
Sar_A	x			FG	sar_A.cif
Sar_B	x			FG	sar_B.cif
opamp	x	x	x	FG	standard.cif
pad				FG	standard.cif
RPA_jumper				FG	standard.cif
resistors	x			P	r.cif
dac_caps	x			P	dac_caps.cif
liblk	x	x	x	P	pliblk.cif nliblk.cif mliblk.cif

Table 4 - AIDE2 library cells and characteristics.

A research effort in cooperation with this one has the objective of developing continuous-time integrators which can be used for implementing active-C filters [15,16]. More information can be found in the project reports [17].

2.5 Application of AIDE2 as a Silicon Compiler

The system designer or novice user who does not wish to write a C language circuit description may design circuits with AIDE2 by the use of a high level synthesis program capable of taking a high level circuit or systems description and translating this description into a C language circuit description which can be input to the AIDE2 program. Once the required cells are placed within the AIDE2 cell library, such a program is not difficult to write.

The objective of the high level program is to take system level specifications and convert them into a circuit description program. Such

programs may have built-in intelligence and may make decisions based upon the information inputted by the designer. These high level programs when interfaced with AIDE2 form an analog silicon compiler for limited classes of analog circuits and systems. AIDE2 has been used to develop a silicon compiler for switched capacitor filters and for successive approximation A/D and D/A converters [5,6].

An area optimized CAD, switched capacitor filter design program called AROMA was developed to interface with an earlier version of AIDE2 designated as AIDE [18]. This high level CAD program contains several user-selectable filter approximation techniques. AROMA permits the user to make tradeoffs between several design parameters such as passive sensitivity, op amp output voltage swings, clock frequency, and the total capacitance of the filter. The program has default values of the tradeoffs, however, each block in the cascade filter can be modified by the user to meet specific requirements. The filter can be specified by one of the following options: 1.) frequency specifications, 2.) continuous-time domain transfer function $H(s)$, and 3.) discrete-time domain transfer function $H(z)$. The output of AROMA is a circuit description program which when applied to AIDE will generate a layout. Further information concerning AROMA can be found in the references [3,6].

A high level design program has been developed which is capable of generating circuit description for AIDE2 of a successive approximation A/D or D/A converter. This high level program has been incorporated into the AIDE2 program to produce a CAD tool capable of high level input and a CIF plot output. Users with little knowledge of analog integrated circuit design can use this program to design and layout successive approximation A/D or D/A converters.

The ADDAC program asks the user for the following input: 1.) the type of

converter (A/D or D/A), the number of bits of resolution, and the required linearity. Based on this and information about the capacitor and resistor ratio accuracies for the particular fabrication process, the program selects the architecture which best meets the required specifications while minimizing the chip area of the converter.

The available architectures use resistors and capacitors to decode the most significant bits and the least significant bits. Which bits are decoded by resistors and which are decoded by capacitors are determined by equations describing the nonlinearity as a function of passive component mismatch. The key aspect of this high level program is the selection of the topology. The variables in this selection algorithm are whether the converter is an A/D or D/A, which bits are decoded by voltage division (resistors), which bits are decoded by charge division (capacitors), and how many bits are decoded by each. The program begins with the initialization of variables used in the program, including values for capacitor ratio accuracy, resistor ratio accuracy, the area associated with a unit capacitor, and the area associated with a unit resistor. These ratio accuracies are process dependent and should be obtained by fabricating the actual capacitor array and resistor string on a test chip for the particular process being used.

The ADDAC program has been used to design an A/D converter which is being implemented on the Harris MPC2 fabrication. This circuit will be used to experimentally characterize the performance of converters designed by ADDAC. The influence of the automated design techniques on the speed and accuracy are of particular interest. The ADDAC program has several areas of improvements that should be considered. These include the reduction of area, consideration of different architectures, and the provision of a simulation input file to a simulator capable of analyzing the performance of an A/D or D/A converter.

2.6 Future Plans for AIDE2

The immediate plans for AIDE2 are to experimentally evaluate the AIDE2 circuits designed and implemented on the Harris MPC2 and on the MOSIS processes. The Harris MPC2 includes both switched capacitor filters and an ADDAC-designed 4-bit A/D converter while the MOSIS circuits include only switched capacitor filters. Plans also include the conversion of the ADDAC cells to the MOSIS technology. The resulting fabrications should permit the evaluation of performance of a given technology as well as the performance of similar circuits implemented in different technologies.

The activities planned for AIDE2 are to investigate the performance of AIDE2-designed circuits and to apply AIDE2 as a platform for research in analog CAD methodology. The first step will be to compare the 5th order, low-pass, filter designed by AIDE2 with an identical custom-designed circuit in industry. The performance comparison will include noise, signal swing, linearity, offset, frequency, and other characteristics. AIDE2 will also be used to design circuits that will be limited because of the inherent circuit performance to see how CAD techniques effect these limitations.

After evaluating the comparison between the performance of CAD designed circuits to custom designed circuits, modifications will be made to AIDE2 to improve the expected performance gap. The increase in performance will come from two sources. One is cells more suitable to CAD and the other is CAD techniques more suitable to cell performance. Potential problems due to CAD techniques include routing parasitics and signal coupling, poor ac grounds due to conductor resistance, insufficient accuracies of passive components, etc. Library cells using circuit techniques less sensitive to CAD induced performance limitation will also be examined.

AIDE2 is presently being evaluated as a CAD tool in several industries.

The primary interest of industry is in the use of AIDE2 as the basis to build application-specific silicon compilers. The feedback expected from this evaluation will provide further direction for development of AIDE2. The Programmers Manual for AIDE2 will be written which will provide more details on the internal aspects of AIDE2.

One of the more important activities of AIDE2 is the development of a cell compiler. This software should allow quick entry of new cells, reduce the dependence on technology, and allow customization of AIDE2 by users. It is important that the cells be area efficient and have the performance necessary to achieve the desired circuit performance. A combination of external software programs and the capabilities of AIDE2 will be used to develop the cell compiler.

3.0 Precision Analog Small Signal Modeling

The accuracy of modeling is a serious problem for analog integrated circuits causing the designer to avoid minimum technological channel lengths causing the designer to avoid minimum feasible channel lengths preventing full utilization of VLSI technologies. One of the major problems in modeling is the accuracy of the small signal model parameters. Because these parameters are based on derivatives or slopes, their accuracy can be very poor although the curve itself may be very accurate. The objective of this research is to develop small signal models with increased accuracy. It is desired to provide small signal conductances and capacitances to within the accuracy which the dc model provides for the large signal model. This typically means small signal models with accuracy of 10% or less.

Since good large signal models have been developed for the dc characteristics of short channel devices, it was decided to use one of these models as the input for the small signal model. The BSIM model developed at

UC Berkeley was selected as the large signal model because of its good performance for short channel devices at normal and subthreshold regions [19]. The BSIM model is widely documented and presently employed in a version of SPICE which made it a very attractive choice for this research activity. The BSIM model solves for the dc operating conditions of the device and also provides a small signal model which is developed by entering the dc information into an analytical expression.

The problem with the BSIM small signal model is shown in Fig. 6. Fig. 6a shows the simulated and experimental comparison of the BSIM large signal model. It is noted that the comparison is quite good with an rms error of 3.37%. Fig. 6b shows the measured and simulated small signal channel conductance from the BSIM model. While the data looks good the rms error is 76%. The reason for this large error is shown in Fig. 6c which shows an expanded version of the saturation region. It is obvious that the BSIM small signal model is not very accurate in the region where it is most often used. The experimental small signal model curves in Figs. 6b and 6c are achieved by differentiating the dc experimental data.

The first step in solving the small signal model problem was to determine what is the experimental value. The approach taken was to experimentally measure the small signal parameters through the use of Fig. 7. In this method the experimental small signal conductances and transconductances are measured directly. It is only necessary to insure that the ac signal swings are small enough to be in the linear range. Comparison of the results of this measurement with the small signal experimental results of BSIM showed excellent agreement. As a consequence, the experimental small signal model parameters achieved by differentiating the dc BSIM experimental data were chosen to be the desired experimental parameters.

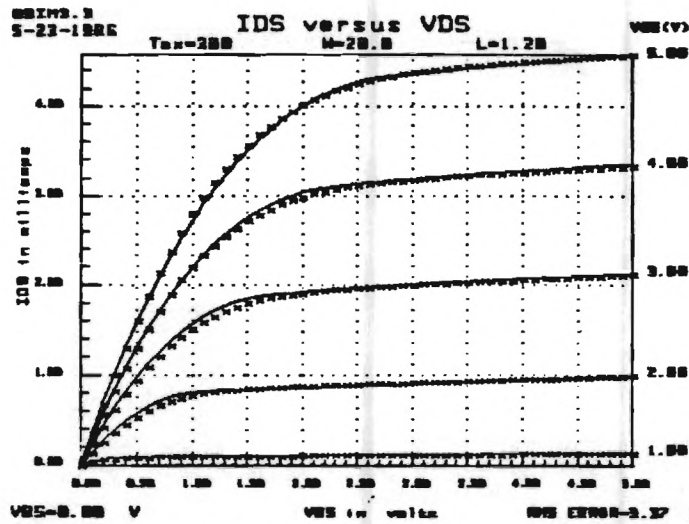


Fig. 6a - Experimental and simulated BSIM large signal model comparison.

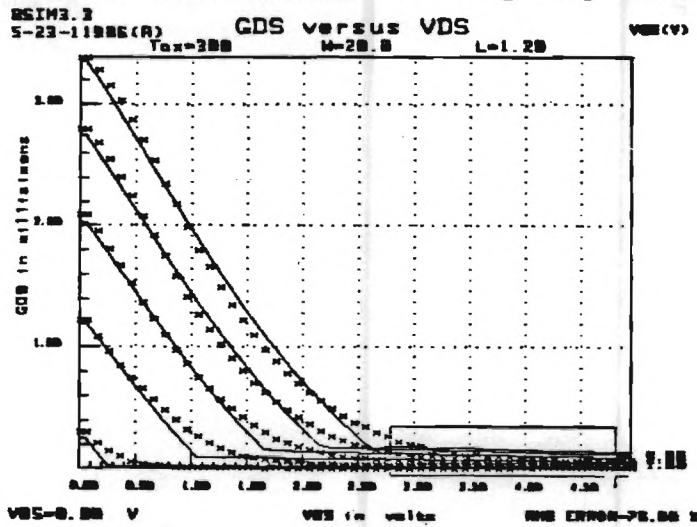


Fig. 6b - Experimental and simulated BSIM small signal model comparison.

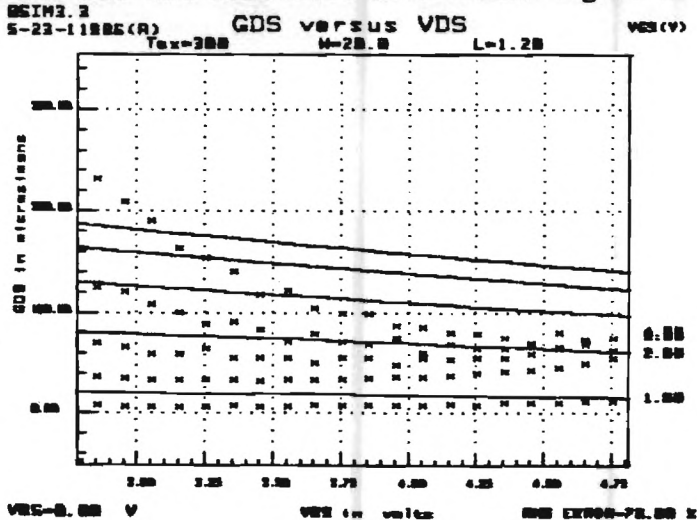


Fig. 6c - Fig. 6b expanded in the saturation region.

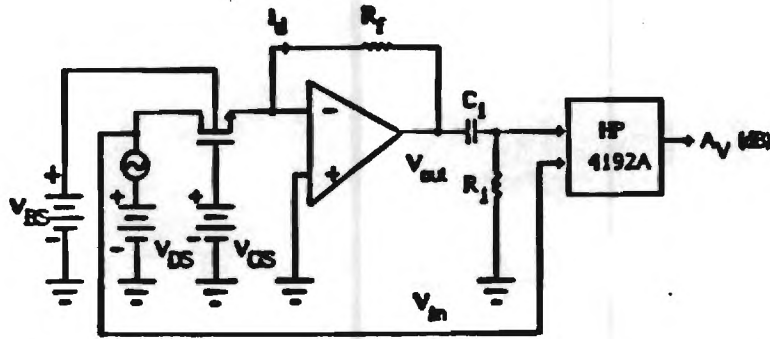


Fig. 7 - Experimental measurement of the small signal model parameters.

The second step was to select a method of developing the small signal model from the BSIM dc experimental data. Three methods were considered to implement the small signal model starting with the dc solution of the BSIM model. These methods are called the analytical, empirical, and table look-up models and their relationship is shown in Table 5. Each approach uses the dc variables provided by the BSIM dc model. The advantages and disadvantages of

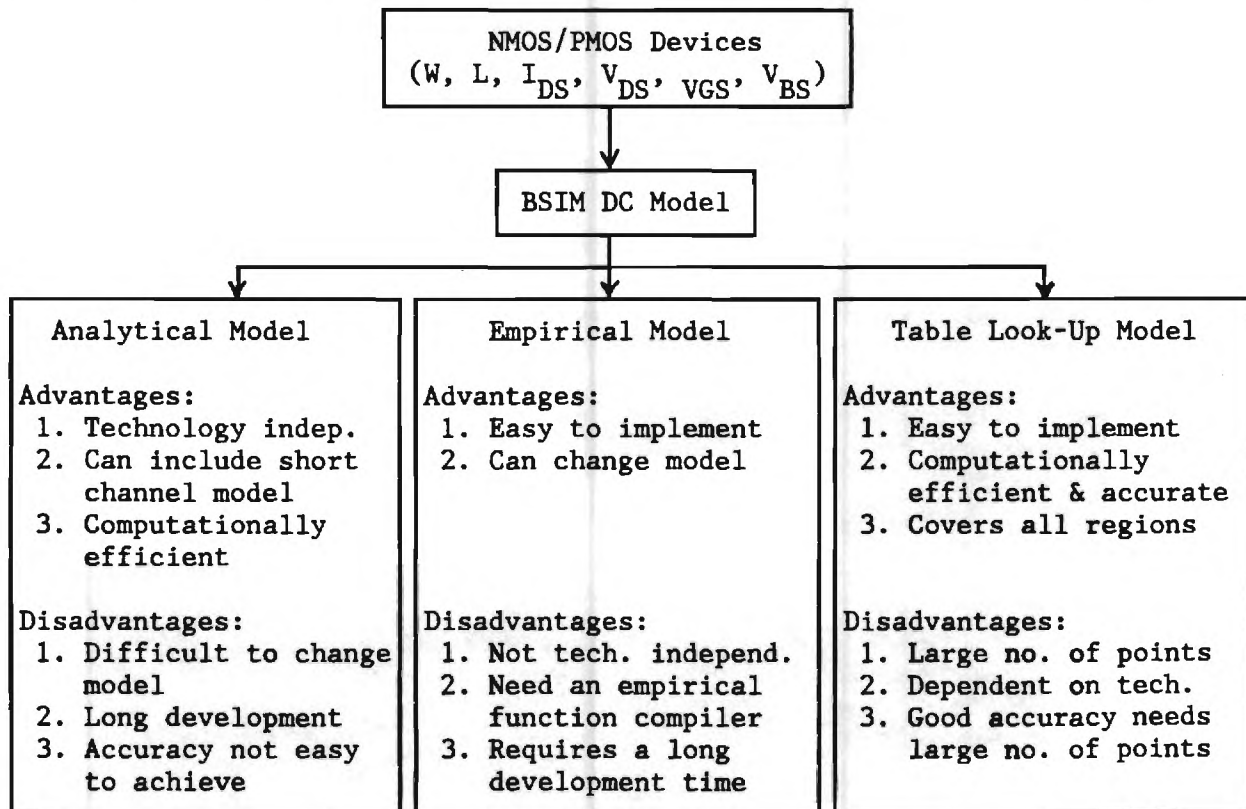


Table 5 - Precision small signal modeling approaches.

each approach are also indicated in the Table. After careful evaluation, a combination of the table look-up and the analytical approach was selected as the approach for the small signal model.

The table look-up approach was applied to the channel conductance parameter, g_{ds} , because it was felt this parameter would be the most difficult to achieve. A routine was developed on the HP 9836 using the HP 4145 Semiconductor Parameter Analyzer which takes the experimental BSIM dc data to generate the channel transconductance as a function of V_{DS} , V_{GS} , W , and L . The number of data points taken are determined by the accuracy desired. Interpolation using the cubic spline technique is used to provide the model value at arbitrary values of V_{DS} , V_{GS} , W , and L . In some cases, analytical functions can be used to assist in the interpolation of the model. 49 data points were required to generate a model for g_{ds} with accuracy of less than 10% for any given value of V_{DS} and V_{GS} for a given W and L . The influence of W and L on g_{ds} are being investigated and will be included into the model.

The advantage of the table look-up model is illustrated in Fig. 8. Fig. 8a shows the experimental and simulated dc characteristics for an NMOS transistor with $W = 20$ microns and $L = 1.5$ microns. Fig. 8b shows the experimental and simulated values for g_{ds} . Fig. 8c shows the expanded part of Fig. 8b in the saturation region. In this case, it is noted that the g_{ds} curves begin to rise as V_{DS} increases which is due to the beginning of voltage breakdown. The table look-up approach models this effect without any extra effort. The model accuracy can be increased by increasing the number of points used to interpolate the model value. An interesting feature of the channel conductance parameter is that it is much harder to measure for long channel devices because the value of g_{ds} is smaller. Correspondingly, the accuracy of the table look-up model becomes worse for longer channel lengths.

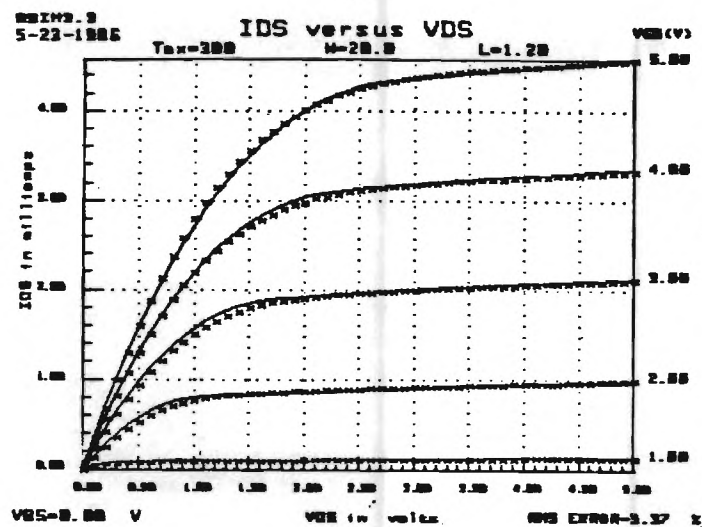


Fig. 8a - Experimental and simulated data for the dc characteristics.

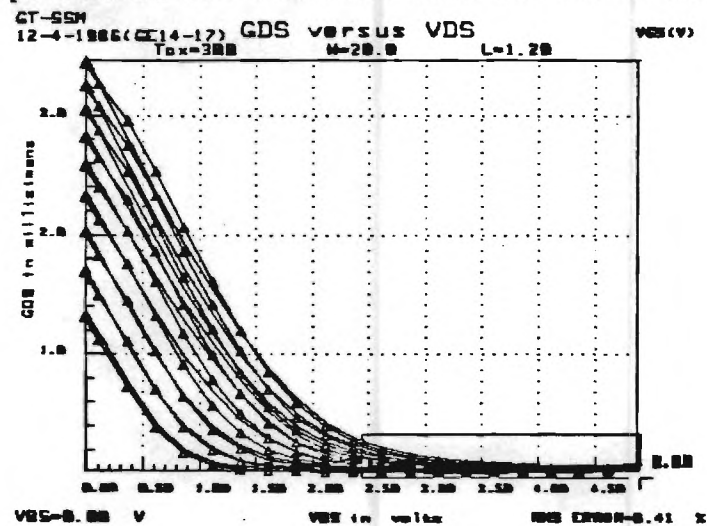


Fig. 8b - Experimental and simulated data for the channel conductance.

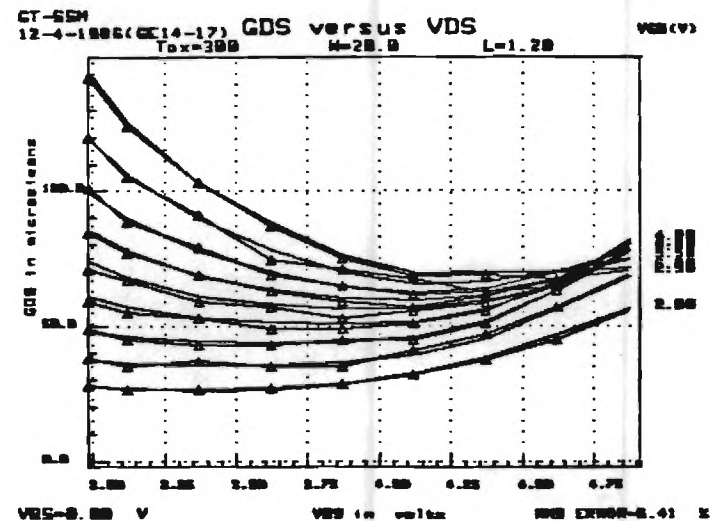


Fig. 8c - Saturation region of Fig. 8b.

Other aspects of this research include the accuracy limits and the application of the model to other devices. If the dc characteristics don't match, the ac characteristics won't match. This is a basic problem for models and simulation which exists no matter the approach. Measurements have been made on different CMOS technologies as well as different devices on the same CMOS technology with acceptable but lesser accuracies observed.

At this point in the research to develop precision small signal models, the solution has been implemented and shown to provide increased accuracy. A table look-up model has been developed for the channel conductance as a function of drain-source and gate-source voltages. The next step is to extend the model to the geometry effects on the channel conductance. At this point, the model techniques should be formalized in the sense of automatically developing a table look-up model which minimizes the number of points and provides the desired accuracy. In general, a small signal model parameter is a function of several variables including device voltages, geometry, and other parameters. A general procedure will be developed which represents the functional dependence of a small signal model on n variables as an n -dimensional table look-up model. The computational speed and memory requirements of this approach will be minimized wherever possible.

The next step in this research is to complete the modeling of the small signal conductances and transconductances. It is expected that the remaining parameters will be easier to model than the channel conductance. It has been shown that the back-gate transconductance is linearly related to the top-gate transconductance [12]. This information will simplify the modeling.

The small signal model capacitances will also be examined in this research. The experimental variation of these capacitors will be determined and compared to present models [20, 21]. If the resulting accuracy is not

satisfactory, the model methodology developed for the transconductances and conductances will be applied to the device capacitors.

Once the small signal model capability has been completed, it will be implemented into a simulator. The PSPICE [22] simulator has the ability to add new models and will probably be used as the simulator. The model extraction program will probably be developed on the HP 9836.

In order to verify the proposed small signal modeling concepts, two op amps will be designed using a 1.2 micron CMOS technology. One op amp will have a device length of 1.6 microns and the other a device length of 3.2 microns. The experimental performance of these op amps will be compared to the simulated results with the purpose of evaluating the capability of the model to accurately predict the small signal performance of a complex analog circuit.

4.0 Multilevel Simulation of Analog Circuits and Systems

The objective of this research is to develop the means for simultaneous, multilevel simulation of analog circuits. Such a capability is necessary in order to be able to simulate analog circuits at the level at which they were designed. Presently, analog circuits are "flattened" to the component level in order to be simulated. While this approach yields a detailed simulation, it does not use computational resources wisely because such detail is normally hidden from the high level design. The best approach is to have a simulator capable of simulating both at high and low levels at the same time.

The simulator under development must be capable of simultaneous simulation at more than one level of circuit complexity. It should permit the time domain simulation of an analog circuit including dc transfer curves and transient response for linear and nonlinear circuits and systems. It should also provide the frequency domain simulation of a linearized circuits. The

levels of complexity include the component level, the circuits level, and the systems level. The simulation primitives of the system level can be rational polynomials, frequency domain, or the time domain. These primitives only model the linear behavior of the system. The simulation primitives for the circuits level is based on macromodels using the component level simulator. These macromodels may be linear or nonlinear. The simulation primitives for the components level are active devices and passive components and are normally nonlinear.

Two basic approaches exist which are suitable for simultaneous, multilevel analog simulation. The first uses combined simulators on the entire circuit and the second partitions the circuits for use by the various simulators. A combination of both approaches will be tried in order to take advantage of the best aspects of each approach. Presently, only the first approach has been implemented. The second approach is under consideration and will be discussed later.

The first approach uses two independent simulators. One simulator is SPICE-PAC [23] and the other is a graphical analysis package called GAP. SPICE-PAC is a segmented version of SPICE which allows internal access to the SPICE program. GAP uses the concepts of signal flow graphs to solve for the transmittance of a circuit represented as a linear graph. The transmittances of each graph can have the form of a rational polynomial in the complex frequency domain of any order. Thus, an entire linear filter can be represented as a single transmittance in the graph. The output of GAP is a rational polynomial in the complex frequency domain with numerical coefficients.

The first approach which uses the SPICE-PAC and GAP simulators uses the following algorithm to analyze the frequency response of a complex analog

system. First, the linear and nonlinear portions of the system are identified. GAP is used for the linear part and SPICE-PAC for the nonlinear part of the system. Obviously, the larger the linear portion of the system, the more efficient this approach will be. The next step is to reduce the linear part of the system to dc or let the frequency variable approach zero. SPICE-PAC is used to solve for the dc operating point of the system. Next, the nonlinear part of the circuit is replaced by its linear graph equivalent and GAP is used to solve for the linear frequency response. This method assumes a relationship between the nonlinear and linear models.

An example of how this algorithm is applied is shown in Fig. 9a. In this analog system, the linear parts are the 4th-order filter, the summer, and the buffer amplifier. The nonlinear part is the active filter shown in Fig. 9b.

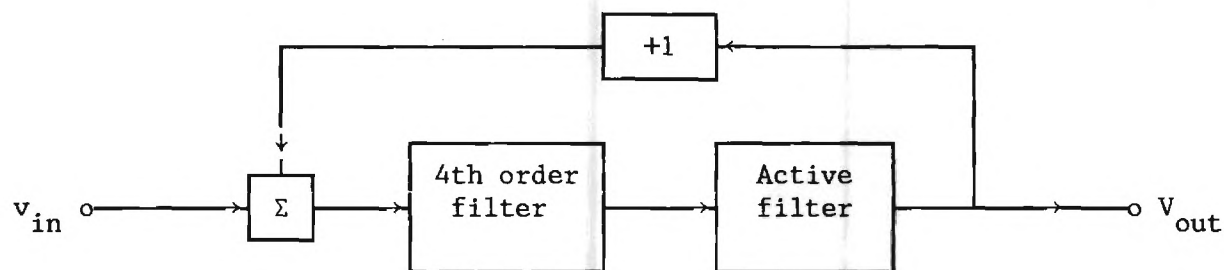


Fig. 9a - Complex analog system.

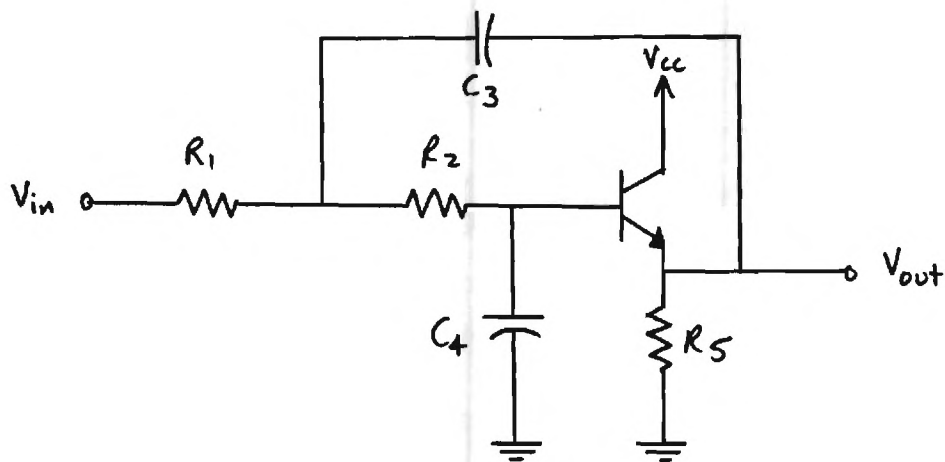


Fig. 9b - Active filter of Fig. 9a.

Fig. 10 shows the simplified graph of the entire analog system. $F_4(s)$ and $F_A(s)$ are the linear transfer functions of the 4th-order filter and active filter, respectively. The first step is to solve for the dc operating point

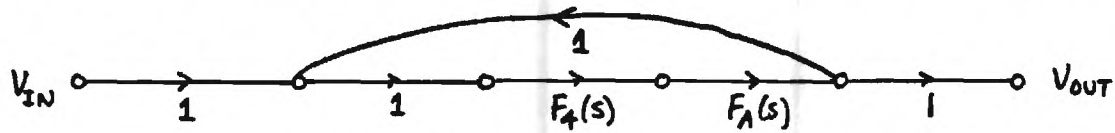


Fig. 10 - Simplified flow graph of Fig. 9a.

of the bipolar transistor using SPICE-PAC. The 4th order filter consists of a frequency independent transfer function, $F_4(0)$ and the capacitors in the active filter are treated as open circuits. Once, the operating point of the bipolar transistor is known, then a linear graph of the active filter can be constructed. This graph is shown in Fig. 11 and includes the linear small signal model of the bipolar transistor. This graph is substituted for the $F_A(s)$ in Fig. 10 and the complete linear transfer function found by GAP.

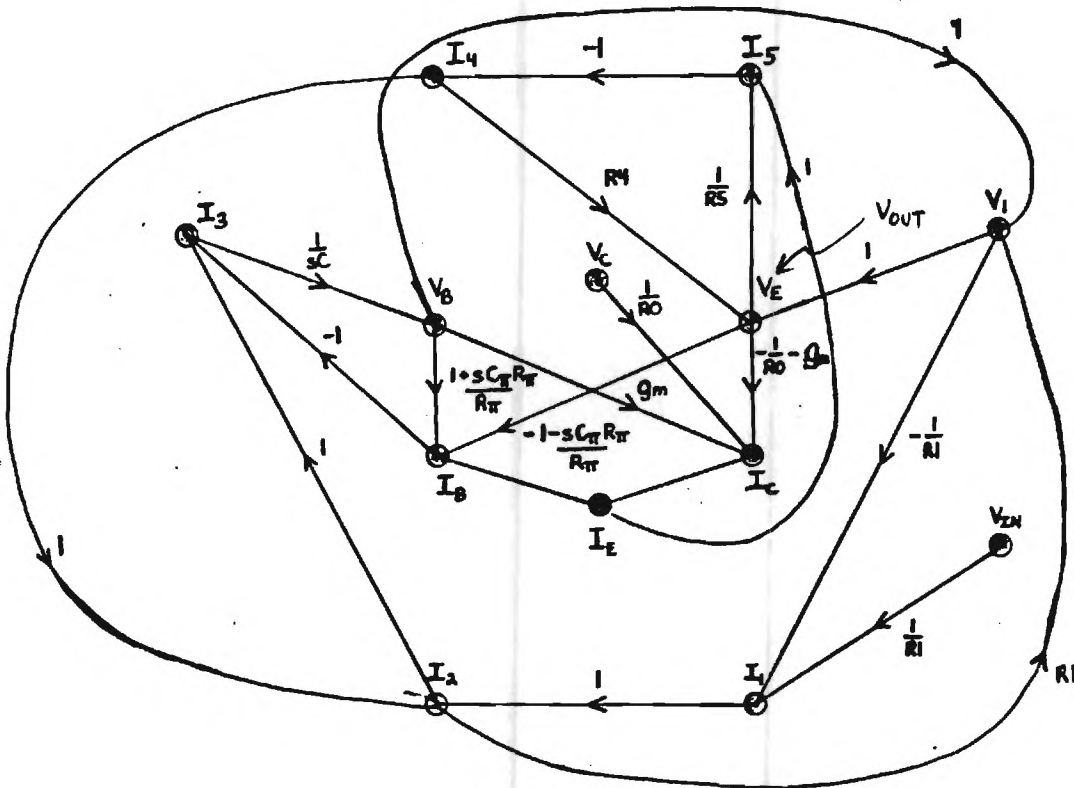


Fig. 11 - Signal flow graph of Fig. 9b.

Because of the signal flow graph simulator efficiency, it is desirable to keep the flow graph to minimum complexity. This can be accomplished by flowgraph simplification techniques before solving the entire circuit. The last step in the simulation is to replace the complex frequency variable by $j\omega$ and plot the magnitude and phase of the frequency response.

This approach has several problems. If the nonlinear part of the circuit is not small, the flow graph becomes complicated and the analysis is not efficient. The model databases must be able to be translated back and forth between simulators. There are essentially two complete models of the analog system. One in the SPICE-PAC format and one in the GAP format. For example, the 4th-order filter of Fig. 9a must be described to GAP as a transmittance of a single branch. However, in SPICE-PAC the 4th-order filter is a voltage-controlled, voltage-source having the gain of the filter at dc. Fig. 12 shows the relationship of the databases in the multilevel analog simulator. A hierarchical component description translator is used to translate between the component and graphical description levels.

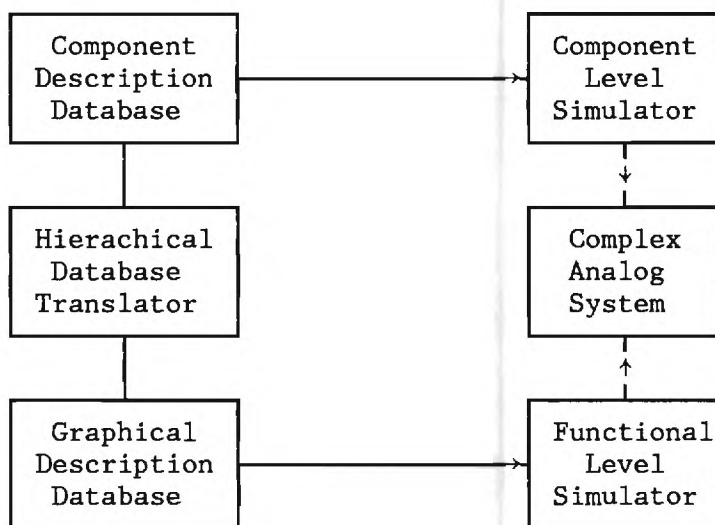


Fig. 12 - Interrelationships of the various databases.

The combined simulator approach can also be used to find the linear time domain response by solving for the poles of the complete frequency response, making a partial fraction expansion, and using the inverse Laplace transform [24]. This method cannot be easily applied to solving the nonlinear time domain response.

The second approach involves partitioning the system into parts. If the network is separated into linear and nonlinear parts, then multilevel simulators can be used to solve the individual parts and the total response developed from the individual responses. Unfortunately, the separation of a complex network is not always straight-forward. Several techniques exist for partitioning a large network. Node tearing identifies the degree of coupling between portions of the network and permits the separation of the network into two or more parts [25]. Another technique is called waveform relaxation and has been used in simulators for both analog and digital networks [26]. These techniques will be examined for their degree of applicability to the simultaneous, multilevel analog simulator. Perhaps a combination of the two approaches may result in the best result.

The present status of this research is the development of the GAP program for frequency domain analysis and the use of SPICE-PAC together with GAP to simulate several analog systems containing linear and nonlinear circuits. The next steps are to automate the interrelationship between GAP and SPICE-PAC as illustrated by Fig. 12. Once this is completed, the GAP program will be extended to include the linear time domain response. The next step is to make a careful comparison of the computer effort to simulate several examples using the normal approach and the multilevel approach. These results will be used to determine the direction of the next step. Areas where the multilevel approach is slow will be identified and solutions investigated. It will also

be necessary to compare the multilevel simulator with the latest developments for low level circuit simulators which include relaxation methods, hardware accelerators, and other techniques. If the multilevel approach still shows promise for the complex class of circuits, then methods of extending the multilevel analysis to the nonlinear frequency domain will be considered.

Other areas of investigation include the examination of macromodeling techniques using SPICE to achieve a level of simulation between the graphical and component levels. The multilevel analog simulation research is expected to provide the first step toward a simulator capable of simulating both analog and digital circuits. The coupling of analog and digital simulators may be more efficient if done at the higher levels of simulation. Because most systems will contain both analog and digital systems in increasing complexity, it is important to lay the groundwork for this very important problem through the present research in analog multilevel simulators.

5.0 Summary and Plans

This report has summarized the results of research in analog CAD methodology during the academic year, 1985-1986. The objective of this research has been to develop methodologies for analog circuit design which can be implemented by the computer. CAD methods have been developed and applied based on the understanding of pertinent design methodologies. These methods have raised analog circuit design to higher levels and have allowed the development of specialized silicon compilers for analog circuits. In addition, this research has focused on the problem of accurate small signal models and higher level simulation techniques for analog circuit design.

5.1 Summary

Previous research developed the CAD platform called AIDE2. This CAD tool has been used to design switched capacitor filters using a parameterized cell

programmed by a description program written in the C-language. Initially, the goal of the AIDE2 program was to be technology independent. This goal was found to be difficult to achieve and instead the technology was fixed on the MOSIS 3 micron, double-poly, CMOS process. This necessitated the redesign of the linear cell used to design the switched capacitor filters. Several new cells have also been added to the AIDE2 library which are used in the design of successive approximation analog-digital converters. These cells were based on the Harris 5 micron CMOS technology but are being converted to the MOSIS 3 micron technology.

Experimental measurements of the circuit designed by AIDE2 and fabricated by Harris uncovered two fatal errors which prevented the circuits from working properly. The first error was a reversal of the op amp inputs and the second was the failure to connect the end of one of the clock inverters in the transfer slice to the power supply. These errors have been corrected and new AIDE2 designs are in fabrication. Included in these designs is a benchmark circuit which has been custom designed in industry on the same technology.

Because AIDE2 has a fixed floorplan, it has been possible to implement a testing capability which guarantees the access of any internal node after fabrication. This capability is normally accessed during partial simulation of the circuit. When the designer makes a partial simulation involving internal nodes, the program provides the option of placing and routing the necessary test cells. A third-order filter has been used to show that a worst case area penalty of 1.76 is required for maximal testing capability.

AIDE2 has been used to develop applications specific silicon compilers. A high-level synthesis program has been written which takes the filter inputs at a high level and automatically creates the description program which is passed to AIDE2. A second example was a high level synthesis program capable

of taking a high level description of an analog-digital or digital-analog converter and providing the input to AIDE2 necessary to automatically layout the converter. Several converters have been designed and are being fabricated.

The impact of the AIDE2 has been in the development of a platform for the investigation of design techniques for analog circuits. The performance of circuits designed by these methods has yet to be carefully compared with custom design circuits. The result will be the identification of methods to improve the use of CAD to design higher performance analog circuits. AIDE2 represents one means by which analog design can be placed in the hands of the systems engineer permitting the successful design of analog integrated circuits.

This research has investigated methods by which precision small signal models can be developed for analog CMOS circuits. It has been found that small signal models used in existing simulators can be off from the experimental value by 40%. In many applications, this error is multiplied and represents a significant cause why the small signal simulation does not predict the experimental small signal response. One result of this research is that good large signal models exist for CMOS devices. The starting point for the precision small signal model is the dc operating point determined by these large signal models. This information is applied to a combination table look-up and analytical model to interpolate values of the small signal model parameters. The accuracy of the small signal model parameters depends upon the number of points stored in the table. It has been found that the accuracy of the small signal model parameters can approach the accuracy of the large signal model for the dc solution.

In the development of AIDE2 and its application to high level design, it

was apparent that a simulation capability for analog circuits was needed at higher design levels. A high level program using graph theory techniques has been written and interfaced with a segmented version of SPICE to simultaneously simulate the frequency response of a circuit with part modeled at the topological level and part modeled at the component level. This simulator has been used to find the frequency response of a complex analog circuit in order to determine its capabilities. The increase of computer efficiency depends on the problem and the number of nonlinearities.

5.2 Plans

The plans for this research activity in analog CAD methodology, modeling, and multilevel simulation are described in the following. Because of the layout errors in the Harris MPC1 fabrication, it is necessary to resubmit the circuits for the next fabrication and to experimentally determine the performance of AIDE2-designed circuits. It should be possible to identify the limitations that CAD techniques place on analog circuits and to develop performance oriented analog CAD tools.

Another important aspect of the AIDE2 development is the development of a cell compiler. The cell compiler provides the means of quickly entering fixed and parameterized cells into the internal library of AIDE2. This compiler is being designed in such a manner as to allow the user to quickly design his own cells using his own technology. In this way, the lack of technology independence is not a serious limitation in the use of the AIDE2 program. The cell compiler will generate the layout of the cell compatible with the AIDE2 floorplan, a simulation file for the cell, and the programming necessary to parameterize the cell. The user starts with a design in mind based on his technology and must enter the technology layer definitions and design rules. Maximum use will be made of existing CAD software internal and external to

AIDE2.

The plans for the precision small signal model are to complete the modeling of the transconductance parameters and to extend the modeling technique to the device capacitors. An important effort will be to increase the model accuracy and at the same time improve the efficiency of the model in regard to computer memory and speed requirements. The next step will be to implement the model into a simulator so that it can be used in the design of analog circuits. Two op amps will be designed and layed out using 1.2 micron CMOS process. One op amp will use channel lengths of 1.6 microns and the other channel lengths of 3.2 microns. The purpose of these op amps is to be able to accurately predict their experimental small signal performance using the modeling technique developed in this research.

The next step in the multilevel analog simulator is to evaluate in more detail its ability to efficiently simulate complex analog systems. The links between the high level and low level simulator were done by hand in the one example that was considered. These links need to be implemented in the software before the program can be evaluated. Another step is to extend the high level simulator to the linear time domain. The final step will be to use the simulator for the nonlinear time domain. Partitioning methods and ways of enhancing the efficiency of the simulator will be examined. Another area that will be considered is the use of macromodeling to achieve an intermediate level based on the low level simulator database. One very important aspect of this research is to lay the groundwork for the development of a simulator capable of simultaneously simulating both digital and analog circuits.

Besides the research described above, methods for automatically designing analog integrated circuits are being examined. In this area a program has been written for the IBM personal computer which takes the specifications for

an op amp and attempts to design a fixed topology, two-stage op amp to meet these specifications. This program is entirely algorithmic and uses no intelligence. It has been used primarily to gain experience. The next step is to incorporate variable topologies and introduce artificial intelligence techniques where appropriate.

6.0 References

- [1] Ralph K. Cavin, III, "1985 Annual Report of SRC Research," Vol. 1, Design Sciences, Semiconductor Research Corporation, P.O. Box 12053, Research Triangle Park, NC 27709, April 1986.
- [2] Phillip E. Allen, "A Computer Aided Design Methodology for Analog LSI/VLSI," SRC Technical Report No. T85014, Semiconductor Research Corporation, P.O. Box 12053, Research Triangle Park, NC 27709, March 1985.
- [3] Phillip E. Allen, "A Multilevel Analog Integrated Circuit CAD Program," SRC Technical Report No. T86045, Semiconductor Research Corporation, P.O. Box 12053, Research Triangle Park, NC 27709, April 1986.
- [4] P. Barton, D. Ellsworth, and S. Hong, "AIDE2.1 User's Guide," School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, June 9, 1986.
- [5] P.E. Allen and P.R. Barton, "A Silicon Compiler for Successive Approximation A/D and D/A Converters," IEEE Custom Integrated Circuits Conference, Rochester, NY, May 1986, pp. 552-555.
- [6] E. Sanchez-Sinencio and J. Ramirez-Angulo, "AROMA: An Area Optimized CAD Program for Cascade SC Filter Design," IEEE Trans. on CAD, Vol. CAD-4, No. 3, July 1985, pp. 296-303.
- [7] D. Cohen and V.C. Tyree, "Quality Control for the Silicon Broker's Perspective," VLSI Design, July/August 1982, pp. 24-30.
- [8] P.E. Allen, H.A. Rafat, S.F. Bily, E.R. Macaluso and A.P. Nedungadi, "The Use of a CAD Tool to Design Analog Integrated Circuits," Proc. of IEEE ISCAS, Montreal, Canada, May 1984, pp. 1223-1226.
- [9] P.E. Allen, E.R. Macaluso, S.F. Bily, and A.P. Nedungadi, "AIDE2: An Automated Analog IC Design System," Proc. of IEEE 1985 Custom IC Conf., Portland, OR, May 1985, pp. 498-501.
- [10] R.W. Hon and C.H. Sequin, "A Guide to LSI Implementation," 2nd Edition, Xero Palo Alto Research Center, Palo Alto, CA, Jan. 1980.
- [11] L.W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," Electronics Research Laboratory, Memorandum No. ERL-520, College of Engineering, University of Calif., Berkeley, CA, May 9, 1975.

- [12] P.E. Allen and D.R. Holberg, CMOS Analog Circuit Design, Holt, Rinehart, and Winston, New York, NY, 1986.
- [13] P. Allen and S. Hong, "Physical Testability of Analog IC's Designed Using AIDE2," Proc. of 29th MWCAS, Lincoln, NB, August 1986.
- [14] User's Manual, MOSIS Implementation System, USC Information Sciences Institute, 4676 Admiralty Way, Marina Del Rey, CA 90292-6695.
- [15] M. Ismail, "A New MOSFET-Capacitor Integrator," IEEE Trans. on Circuits and Systems, Vol. CAS-32, PP. 1194-1196, Nov. 1985.
- [16] Y.P. Tsividis, et. al., "MOSFET-C Filters in VLSI," IEEE J. of Solid-State Circuits, pp. 15-30, Feb. 1986.
- [17] M. Ismail, Quarterly Progress and Annual Reports, Semiconductor Research Corporation, P. O. Box 12053, Research Triangle Park, NC 22709.
- [18] P.E. Allen, H.A. Rafat, E.R. Macaluso, and A.P. Nedungadi, "AIDE - Analog Integrated Circuit Design Enhancement Program," Proc. of 26th MWCAS, Puebla, Mexico, August 1983, pp. 622-626.
- [19] B.J. Sheu, D.L. Scharfetter and H.C. Poon, "Compact Short-Channel IGFET Model (CSIM)," and B.J. Sheu, D.L. Scharfetter, C. Hu and D.O. Pederson, "A Compact IGFET Charge Model," Memorandum No. UCB/ERL M84/20, Electronics Research Laboratory, College of Engineering, Univ. of Calif., Berkeley, CA 94720, March 12, 1984.
- [20] A. Valdimirescu and S. Liu, "The Simulation of MOS Integrated Circuits Using SPICE2," Memorandum UCB/ERL M80/7, Electronics Research Laboratory, College of Engineering, Univ. of Calif., Berkeley, CA 94720, Oct. 1980.
- [21] J. J. Paulos and D. A. Antoniadis, "Measurement of Minimum-Geometry MOS Transistor Capacitances," IEEE J. of Solid-State Circuits, Vol. SC-20, pp. 277-283, Feb. 1985.
- [22] PSPICE Manual, MicroSim Corporation, 23175 La Cadena Dr., Laguna Hills, CA 92653, October 1986.
- [23] W.M. Zuberek, SPICE-PAC 2Ga.84.05 User's Guide, Dept. of Computer Science, Memorial University of New Foundland, St. John's, New Foundland, Canada, A1C 5S7.
- [24] L.P. Huelzman, Digital Computations in Basic Circuit Theory, Mc Graw-Hill book Co., New York, NY, 1968.
- [25] N.B.G. Rabbat, A.L. Sangiovanni-Vincentelli, and H.Y Hsieh, "A Multilevel Newton algorithm with macromodeling and latency for the analysis of large-scale nonlinear circuits in the time domain," IEEE Trans. on Circuits and Systems, Vol. CAS-26, no. 9, pp. 735-741, September 1979.
- [26] A.R. Newton and A.L. Sangiovanni-Vincentelli, "Relaxation-based Electrical Simulation," IEEE Trans. on Electron Devices, vol. ED-30, no. 9, pp. 1184-1207, Sept. 1983.